应用型高等院校经管类系列实验教材 · 数学

# MATLAB软件与数学实验

MATLAB Ruan Jian Yu Shu Xue Shi Yan

胡 蓉／主编



经济科学出版社
Economic Science Press

ESP

# MATLAB软件与数学实验

MATLAB Ruan Jian Yu Shu Xue Shi Yan

胡 蓉/主编

封面设计：王 坦




ISBN 978-7-5058-9734-2
定 价：22.00元

应用型高等院校经管类系列实验教材·数学

# MATLAB软件与数学实验

*MATLAB RuanJian Yu ShuXue ShiYan*

胡蓉／主编



经济科学出版社
Economic Science Press




**图书在版编目（CIP）数据**

MATLAB软件与数学实验／胡蓉主编．—北京：经济科学出版社，2010.8

（应用型高等院校经管类系列实验教材·数学）

ISBN 978-7-5058-9734-2

Ⅰ.M… Ⅱ.①胡… Ⅲ.①高等数学—实验—计算机辅助计算—软件包，MATLAB—高等学校—教材 Ⅳ.①013-33 ②0245

中国版本图书馆CIP数据核字（2010）第145669号

责任编辑：白留杰　董海峻
责任校对：杨　海
技术编辑：李长建

**MATLAB软件与数学实验**

胡　蓉　主编

经济科学出版社出版、发行　新华书店经销

社址：北京市海淀区阜成路甲28号　邮编：100142

教材编辑中心电话：88191354　发行部电话：88191540

网址：www.esp.com.cn

电子邮件：bailiujie518@126.com

北京汉德鼎有限公司印刷

季峰装订厂装订

787×1092　16开　12印张　280000字

2010年8月第1版　2010年8月第1次印刷

ISBN 978-7-5058-9734-2　定价：22.00元



# 总　序

实践教学是高等教育本质的必然要求，是践行应用型人才培养的必经之路，是地方行业性教学型本科院校办学的重要特征。近几年来，各高校经济与管理类专业实验教学已经逐步开展，把实验教学作为教学改革的抓手、知识融合的平台以及联系社会的桥梁，然而如何进一步完善实验教学体系、提高实验实践教学水平与质量已经成为各高校亟待解决的问题。应用型高等院校经管类系列实验教材以提高高等院校经济与管理类专业实验教学的建设水平为目的，以实验教材建设为突破口，探讨高等院校经济与管理类实验教材的新方向、新思路、新内容、新模式。

本系列实验教材的编写紧紧围绕“知行合一，能力为尚，积淀特色，共享协作”的地方行业性教学型经济与管理类实验教学理念，贯彻以现代教育技术为基本手段，以实验资源共享与应用为条件，强化理论教学与实践教学互动与互补，“实践与理论相结合”和在“做中学”的指导思想，强调实验教材建设与实验课程建设、实验项目建设、实验教师队伍建设以及深化实验教学改革相结合，力图通过系列教材建设规范实验教学内容和实验项目，促进实验教学质量的提高。

（一）本系列实验教材内容与教学方式符合实验教学规律和要求。具体表现在以下几个方面：

1. 实验教材以实验项目为章节，按如下体例编写：实验目的和实验要求；实验的基本原理；实验仪器、软件和材料或实验环境；实验方法和操作步骤；实验注意事项；数据处理和实验结果分析；实验报告。当然，对于不同的课程，根据其本身的学科特点，实验教材的编写体例并不完全一致。

2. 增加综合性、设计性、创新性实验项目的比例，并逐步将科研成果项目转化为教材的实验项目。

3. 与当前流行的实验平台软件或硬件及教材内容紧密结合，符合一般软件要求。

4. 充分体现以学生为主体，明确实验教学的内涵。实验教学过程体现以学生操作为主，教师辅导为辅，少量时间教师讲解，大部分时间学生操作的特点。

5. 按实验教学规律分配学时，并且有多余的实验项目供学生利用开放实验室自主学习。

6. 内容精练，主次分明，详略得当，文字通俗易懂，图表与正文密切配合。

（二）本系列实验教材遵循实验教学规律，体现时代特色，总体来说，具有以下四个特点：

1. 与现代典型案例相结合。以培养应用型人才为原则，根据实验教学大纲，注重理论联系实际，教材具有较强的实践性、新颖性、启发性和适用性，有利于培养学生的实践能力和创新能力。

2. 建设形式新颖。实验教材分为纸质实验教材和网络资源的形式；纸质教材实验报告



尝试做成活页形式，或做成可撕下的带切割线形式；在纸质教材出版，配套建有供学生实验前和实验后学习使用的网络资源。

3. 实验内容创新。对于实验教材编写内容上的创新，一是凸显应用型人才培养特色实验项目，提高了综合性、设计性、创新性实验项目的比例；二是将教师的科研成果转化为本科学生实验教学项目。

4. 编写程序严格。对实验教材的申请立项的实验教材经由学院领导及专家进行立项审查；实验教材初稿经由相关同行专家给出鉴定，最终审核后，送交出版社评审出版。

本系列教材得到各方面人士的指导、支持和帮助，尤其是得到中国经济信息学会实验经济学与经济管理实验室专业委员会的专家，广东金电集团等多家业界人士，以及各高校同行老师们的支持和帮助，我们在此表示由衷的感谢。本系列实验教材尚处于探索阶段，作为一种努力和尝试，存在诸多不足之处，竭诚希望得到广大同行及相关专家的批评指正。

**应用型高等院校经管类系列实验教材编委会**

2009 年 12 月



# 前　言

近年来，MATLAB 作为一种广泛应用于工程计算及数值分析领域软件工具，越来越受到国际科学界的重视。MATLAB 具有强大的科学计算功能，它的各种工具箱能实现在线性代数、数值分析、数理统计、自动控制理论、数字信号处理、时间序列分析、动态系统仿真、图像处理、生物信息处理、金融信息处理等科学及工程领域的各种计算。因此，MATLAB 已被很多高校列为理工科学生的必修课。

作为金融财经类专业的学生，具有良好的数学基础和能熟练掌握一种数据计算和分析软件是十分必要的。借助 MATLAB 强大的运算平台及其数据处理功能，能够对一些经济学、金融学理论进行模拟，对一些实际数据进行实证，这些实践过程能够加深学生对理论的理解，同时也锻炼了学生的动手能力，提高了分析问题和解决问题的能力。因此，为金融财经类专业的学生开设 MATLAB 软件课是符合学科发展趋势的。

MATLAB 作为一种高级的计算机软件和程序语言，必须通过大量的上机实验才能学好它，掌握它。实验的内容决定了能否达到培养学生动手能力的目的。目前，国内关于 MATLAB 的实验教材很多，实验内容的设置基本分两类，第一类是侧重软件实验。这类教材主要内容为 MATLAB 的基本操作和各种运算功能的实现。第二类是侧重数学实验。这类教材的主要内容为利用 MATLAB 作为平台进行各种数学计算方法的实验。能把两种实验内容结合起来的实验教材不多。另外，要加强学生对实际问题的分析和解决能力，就应该选取一些具体的实例作为实验内容，这有助于学生对理论应用方法和步骤的理解和掌握。因此，我们编写了这本《MATLAB 软件与数学实验》，作为 MATLAB 理论课的配套实验课教材。

本实验教材分为三个部分。第一部分为 MATLAB 简介。因为理论课教材对 MATLAB 各种功能有更详细介绍，所以在这个部分我们仅仅对 MATLAB 基本操作和相关计算函数做了一个概括性的介绍，以便学生在进行实验前对相关理论知识做一个回顾。第二部分为 MATLAB 软件实验。这部分是作为 MATLAB 软件理论课的课程实验的配套实验，属于软件实验部分。该部分共五个实验，10 个学时左右。这部分实验主要是让学生掌握 MATLAB 语言的基本操作和语法，熟悉 MATLAB 的编程环境，侧重于对软件的熟练运用。第三部分为数学实验。这部分实验是作为独立实验课的配套实验内容，属于应用实验部分。该部分共九个实验，实验六至实验十二为单项实验，实验十三和实验十四为综合实验，共 36 学时。这部分的实验主要是训练学生应用 MATLAB 解决各种数学计算问题。在这部分的实验中，我们首先给出学生相关问题的数学模型，学生直接应用 MATLAB 求解，当学生掌握了求解方法后，我们给出应用实例，要求学生首先对实际问题进行分析，建立数学模型，找到求解的方法，再应用 MATLAB 求解。这样的实验内容的设置，有利于学生对实际问题处理方法的掌握。最后两个综合实验都是具体的实际问题，通过前七个实验的训练，要求学生能独立完成对问题的分析、建模和求解的全过程。



本教材的每个实验中都配有相应的实验报告。实验报告中每一项实验都有详细的实验步骤和过程，引导学生一步一步完成实验。我们还为学有余力的同学安排了选做内容，可作为学生课外实验和自我提高之用。

本教材适用于开设 MATLAB 软件课和数学实验课的本科生作为实验教材使用。考虑到各专业对数学基础的要求不同及学时数不等的情况，指导教师可以根据授课专业的不同进行调整，选做其中某一些实验。

本教材由胡蓉担任主编，负责全书的设计和编撰。本教材在设计的过程中得到了张学奇教授的指点和帮助，在编写过程中，廖文辉老师和骆世广老师也提出了很多宝贵的意见和建议，在此，表示衷心的感谢。由于水平有限加上时间仓促，本教材难免会有不妥或失误之处，恳请同行和读者批评指正。

**编　者**



# 目 录

## 第一部分 MATLAB 简介



## 第二部分 MATLAB 软件实验



## 第三部分 数学实验





# 第一部分

# MATLAB 简介

MATLAB 是 Matrix Laboratory 的缩写，意为“矩阵实验室”，是一个集数值计算、符号分析、图像显示、文字处理于一体的大型集成化软件。它最初由美国的 Cleve Moler 博士研制，其目的是为线性代数等课程中的矩阵运算提供一种方便可行的实验手段。经过多年的市场竞争和发展，MATLAB 已发展成为在自动控制、生物医学工程、信号分析处理、语言处理、图像信号处理、雷达工程、统计分析、计算机技术、金融界和数学界等各行各业中都有极其广泛应用的数学软件。

MATLAB 是一个交互式的系统，它的基本运算单元是无需指定维数的矩阵，按照 IEEE 的数值计算标准（能正确处理无穷数 Inf（Infinity）、无定义数 NaN（Not-a-number）及其运算）进行计算。系统提供了大量的矩阵及其他运算函数，可以方便地进行一些很复杂的计算，而且运算效率极高。MATLAB 命令和数学中的符号、公式非常接近，可读性强，容易掌握，还可利用它所提供的编程语言进行编程，完成特定的工作。除基本部分外，MATLAB 还根据各专门领域中的特殊需要提供了许多可选的工具箱，使其应用范围更广，使用更方便。

归纳起来，MATLAB 具有以下几个特点：易学、适用范围广、功能强、开放性强、网络资源丰富。由于 MATLAB 的强大功能，它能使使用者从繁重的计算工作中解脱出来，把精力集中于研究、设计以及基本理论的理解上，所以，MATLAB 已成为在校大学生、硕士生、博士生所热衷的一款计算软件。在此，我们将 MATLAB 中有关数学计算的一些基本功能作一个简单的介绍，为后面的实验部分做好准备。MATLAB 有各种版本，早期有 MATLAB 1.0 for 386 的 DOS 版本，后来逐步发展。本书将以 MATLAB7.0 作为讲解的版本。

第一章

# MATLAB的安装与启动

## 一、MATLAB的安装

放入MATLAB安装光盘，按照安装向导进行安装。在安装过程中可对需要安装的MATLAB组件进行选择，通过“勾选”决定哪些组件被安装。其中有些组件是必须装的，有些是可选的（见表1-1-1）。

**表1-1-1　　MATLAB部分安装组件**

| 组件名称 | 功　　用 |
| --- | --- |
| 必须选择的本原性组件 | |
| MATLAB | 这是最核心的部分。没有它就没有MATLAB环境。有了它，就可以对各类数据（除符号类数据以外）进行操作、运算和可视化 |
| 最常选的通用性工具包组件 | |
| Symbolic Math Toolbox | 符号类数据的操作和计算 |
| Extended Symbolic Math Toolbox | 更丰富的符号计算函数和指令 |
| 其他通用性工具包组件 | |
| Simulink | 不用编写程序，利用方块图实现建模和仿真；主要用于研究微分和差分方程描写的非线性动态系统 |
| Optimization Toolbox | 包含求函数零点、极值、规划等优化程序 |
| MATLAB Compiler | 把MATLAB的M文件编译成独立应用程序 |
| MATLAB *builder for* Excel | 与MATLAB *Compiler*配合使用，生成Excel插件 |
| 常用专业性工具包组件 | |
| Control System Toolbox | MATLAB涉及控制的工具包有10多个，而本栏工具包是最基本的，是解决控制问题所必选的。至于其他控制工具包则是用户根据需要选择的 |
| Signal Processing Toolbox | 是MATLAB信号处理中的基本工具包 |
| Spline Toolbox | 内含样条和插值函数 |
| Statistics Toolbox | 包含进行复杂统计分析所需的程序 |
| 其他专业性工具包组件（举例） | |
| Stateflow | 与Simulink配合使用，主要用于较大型、复杂的（离散事件）动态系统的建模、分析、仿真 |
| System Identification Toolbox | 动态系统辨识 |
| …… | |



## 二、启动 MATLAB

点击桌面的 MATLAB 图标，或从“开始”菜单进入 MATLAB 程序，MATLAB 命令窗（MATLAB Command Window）在一段提示信息后，出现系统提示符“ >> ”。在提示符后，可以输入命令、编程、进行计算。

## 三、学会使用 help 命令

在命令窗内输入 help 命令，再敲回车键。在屏幕上将出现在线帮助总览。MATLAB 命令被输入后，必须敲回车键才能执行，下文不再每次提醒“敲回车键”。学会使用 help 命令，是学习 MATLAB 的有效方法。例如：在“matlab \ lang”（总览的第三行）可以了解到程序语言的结构，在“matlab \ elmat”中可以查到矩阵操作命令。如果想知道 MATLAB 中的基本数学函数有哪些，可以在“matlab \ elfun”中查到。在 MATLAB 中，“基本数学函数”用 elfun 表示，可进一步键入：“help elfun”，屏幕上将出现“基本数学函数”表。如果想了解某个函数的用法，可键入：“help 函数名称”（help 命令与函数名称之间应加空格）。例如：想了解 sin 函数怎样使用，可键入“help sin”。在工具栏中点击 help 按钮，或点击?号按钮，与上面获取帮助信息的方法是等效的。

## 四、学会使用 Demo 命令

在命令窗内输入 Demo 命令，屏幕上将出现演示窗口。整个界面分为左右两个主要的窗口，左边的窗口显示欲演示内容的大标题，点击标题前面的加号，可展开其中的内容。选定其中一项，右边的窗口显示关于大标题的一些说明及一些具体的演示例子。可以直接在右边的窗口看，也可以点击右上角的“Run in the Command Window”，可以在命令窗口运行程序。点击左上角可查看该演示的 M 文件。仔细研究演示程序的 M 文件，也是学习 MATLAB 的一种有效方法。

进入演示窗还有另一方法：在工具栏中点击 Help 栏，下拉式菜单中点击 Demos 项，即可进入演示窗口。

## 五、退出

在工具栏中点击 File 按钮，在下拉式菜单中单击 Exit MATLAB 项，或直接点击界面右上方“关闭”按钮，或在命令窗口中输入“exit”命令，退出 MATLAB。

# 第二章

# MATLAB 基本用法

## 一、变量

在 MATLAB 中，变量由英文字母、数字和下划线组成。第一个字符必须是字母。一个变量最多由 63 个字符组成，并区分大小写。表 1－2－1 是 MATLAB 中表示特殊量的字符。

**表 1－2－1　　MATLAB 中表示特殊量的字符**

| 特殊的变量、常量 | 取　值 |
|---|---|
| ans | 用于结果的缺省变量名 |
| pi | 圆周率 π 的近似值（3.1416） |
| eps | 数学中无穷小（epsilon）的近似值（2.2204e－16） |
| inf | 无穷大，如 1/0＝inf（infinity） |
| NaN | 非数，如 0/0＝NaN（Not a Number），inf / inf＝NaN |
| i，j | 虚数单位：$i = j = \sqrt{-1}$ |

## 二、语句

MATLAB 语句的一般形式为：变量＝表达式。当某一语句的输入完成后，按回车键，计算机就执行该命令。表达式由操作符或其他特殊字符、函数和变量名组成。表达式的结果为一个矩阵，显示在屏幕上，同时保存在变量中以留用。如果该语句末没输入其他符号或输入了逗号，将显示执行结果；如果句末输入了分号，将不显示执行结果，这对有大量输出数据的程序特别有用。如果语句中省略了变量和等号，那么计算机将结果赋值给默认的变量 ans。

如果表达式很长，一行放不下，可以键入“…”（三个点，但前面必须有个空格，目的是避免将形如“数 2 …”理解为“数 2.”与“..”的连接，从而导致错误），然后回车。



## 三、数和算术表达式

MATLAB 中数的表示方法和一般的编程语言没有区别。如：

| | | |
|---|---|---|
| 5 | -78 | 0.0018 |
| 4.86252 | 1.6021E-20 | 9.02252e23 |

在计算中使用 IEEE 浮点算法其舍入误差是 eps，大约保持有效数字 16 位，浮点数表示范围是 $10^{-308}\sim10^{308}$（见表 1-2-2）。

**表 1-2-2　　数学运算符**

| | 数学表达式 | 矩阵运算符 | 数组运算符 |
|---|---|---|---|
| 加 | a+b | a+b | a+b |
| 减 | a-b | a-b | a-b |
| 乘 | a×b | a*b | a.*b |
| 除 | a÷b | a/b 或 b\ a | a./b 或 b.\ a |
| 幂 | $a^b$ | a^b | a.^b |

这里 1/5 和 5\1 有相同的值，都等于 0.2（注意比较：1\5=5）。只有在矩阵的除法时左除和右除才有区别。

## 四、命令窗口的相关操作

1. 命令窗口的常用控制命令。

clf：清除图形窗。

clc：清除指令窗中显示内容。

clear：清除 MATLAB 工作空间中保存的变量。

edit：打开 M 文件编辑器。

who：列出 Work Space 中驻留的变量清单。

whos：列出 Work Space 中驻留的变量清单及变量属性。

save：保存 Work Space 中的变量。

load：加载指定变量到 Work Space 中。

md：创建目录。

cd：设置当前工作目录。

↑（↓）：向前（或向后）调出已输入过的命令。

exit：关闭/退出 MATLAB。

2. 命令行中的标点符号。

命令行中有很多符号具有不同的作用，如表 1-2-3 所示。



**表 1-2-3　　命令行中的标点符号**

| 名称 | 标点 | 作　用 |
| --- | --- | --- |
| 空格 | | （为机器辨认）用作输入量与输入量之间的分隔符 |
| 逗号 | , | 用作要显示计算结果的指令与其后指令的分隔；用作输入量与输入量之间的分隔符；用作数组元素分隔符号 |
| 分号 | ; | 用作不显示计算结果指令的“结尾”标志；用作不显示计算结果指令与其后指令的分隔；用作数组的行间分隔符； |
| 冒号 | : | 用以生成一维数组；用作单下标援引时，表示全部元素构成的长列；用作多下标援引时，表示那维上的全部 |
| 注释号 | % | 注释语句标识 |
| 单引号对 | ' ' | 字符串记述符 |
| 圆括号 | ( ) | 在数组援引时用；函数指令输入变量列表时用 |
| 方括号 | [ ] | 输入数组时用；函数指令输出变量列表时用 |
| 续行号 | … | 由三个以上连续黑点构成。它把其下的物理行看作该行的“逻辑”继续，以构成一个较长的完整指令 |
| “At”号 | @ | 放在函数名前，形成函数句柄；放在目录名前，形成用户对象类目录 |

注：为保证指令正确执行，以上符号一定要在英文状态下输入

3. 输出格式。

任何 MATLAB 语句执行结果都可在屏幕上显示，同时赋给指定的变量，没有指定变量时赋给 ans，数字显示格式可由 format 命令来控制（Windows 系统下的 MATLAB 系统的数字显示格式可以由 Option 菜单中的 Numerical Format 菜单改变）。format 仅影响矩阵的显示，不影响矩阵的计算与存储。MATLAB 以双精度执行所有的运算。

首先，如果矩阵元素是整数，则矩阵显示就没有小数，如 x = [ -1　0　1]，结果为：

```
x =
     -1   0   1
```

如果矩阵元素不是整数则输出形式可由“format 格式”的命令形式进行切换。具体格式命令和说明如表 1-2-4 所示。

**表 1-2-4　　输出格式命令**

| 格式 | 中文解释 | 说　明 |
| --- | --- | --- |
| format | 短格式（缺省格式） | Default. Same as SHORT |
| format short | 短格式（缺省格式） | Scaled fixed point format with 5 digits（只显示五位十进制数） |
| format long | 长格式 | Scaled fixed point format with 15 digits |
| format short e | 短格式 e 方式 | Floating point format with 5 digits |
| format long e | 长格式 e 方式 | Floating point format with 15 digits |
| format short g | 短格式 g 方式 | Best of fixed or floating point format with 5 digits |

| 格式 | 中文解释 | 说　明 |
|---|---|---|
| format long g | 长格式 g 方式 | Best of fixed or floating point format with 15 digits |
| format hex | 16 进制格式 | Hexadecimal format |
| format + | +格式 | The symbols +, − and blank are printed for positive, negative and zero elements. Imaginary parts are ignored |
| format bank | 银行格式 | Fixed format for dollars and cents |
| format rat | 有理数格式 | Approximation by ratio of small integers |
| format compact | 压缩格式 | Suppress extra line-feeds |
| format loose | 自由格式 | Puts the extra line-feeds back in |

例如：

x = [4/3　1.2345e−6]

在不同的输出格式下的结果为：

| | | |
|---|---|---|
| 短格式 | 1.3333 | 0.0000 |
| 短格式 e 方式 | 1.3333e+000 | 1.234e−006 |
| 长格式 | 1.333333333333333 | 0.000001234500000 |
| 长格式 e 方式 | 1.333333333333333e−000 | 1.23450000000000e−006 |
| 有理数格式 | 4/3 | 1/810045 |
| 16 进制格式 | 3ff5555555555555 | 3eb4b6231abfd271 |
| +格式 | + | + |

对于短格式，如果矩阵的最大元素比数 999999999 大，或者比数 0.0001 小，则在打印时，将加入一个普通的长度因数。如 y = 1.e20 * x，意为 x 被 $10^{20}$乘，结果为：

y =

　1.0e+020 *

　　1.3333　　0.0000

“+”格式是显示大矩阵的一种紧凑方法，“+”，“−”和空格显示正数、负数和零元素。

最后，format compact 命令压缩显示的矩阵，以允许更多的信息显示在屏幕上。

书

# 第三章

# 向量与矩阵运算

MATLAB 能处理数、向量和矩阵。但事实上一个数可看成一个 $1\times1$ 的矩阵，1 个 n 维向量也不过是一个 $1\times n$ 或 $n\times1$ 的矩阵。从这个角度上来讲，MATLAB 处理的所有的数据都是矩阵。MATLAB 的矩阵处理能力是非常灵活、强大的。以下我们将从矩阵的产生、基本运算、矩阵函数等几个方面来说明。

## 一、向量及矩阵的生成

1. 直接输入法。

对于 m 行 n 列的矩阵 A，设 $A_{m\times n}=\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \vdots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}$，则在 MATLAB 中矩阵的输入方法如下：A = [$a_{11}$，…，$a_{1n}$；…；$a_{m1}$，…，$a_{mn}$]。矩阵用方括号括起，元素之间用空格或逗号分隔，矩阵行与行之间用分号分开。例如：

输入：

```
A = [1  2  3;  4  5  6;  7  8  9]
```

系统输出：

```
A =
    1  2  3
    4  5  6
    7  8  9
```

表示系统已经接收并处理了命令，在当前工作区内建立了矩阵 A。

2. 利用“:”运算符产生向量。

除了直接法外，最常用的用来产生相同增量的向量的方法是利用“:”算符。在 MATLAB 中，它是一个很重要的字符。例如：

```
z = 1:6
z =
```



1  2  3  4  5  6

即产生一个 1 ~6 的单位增量是 1 的行向量，此为默认情况。

用“:”号也可以产生单位增量不等于 1 的行向量，语法是把增量放在起始点和终止点的中间。例如：

x =0: pi/4: pi

即产生一个由 0 ~ pi 的行向量，单位增量是 pi/4 =3. 1416/4 =0. 7854。

x =

0  0. 7854  1. 5708  2. 3562  3. 1416

也可以产生单位增量为负数的行向量。例如：

y =6: -1: 1

y =

6  5  4  3  2  1

3. 利用函数产生。

常见的矩阵生成函数如表 1 -3 -1 所示。

**表 1 -3 -1　　常见矩阵生成函数**

| | |
|---|---|
| zeros（m，n） | 生成一个 m 行 n 列的零矩阵，m = n 时可简写为 zeros（n） |
| ones（m，n） | 生成一个 m 行 n 列的元素全为 1 的矩阵，m = n 时可写为 ones（n） |
| eye（m，n） | 生成一个主对角线全为 1 的 m 行 n 列矩阵，m = n 时可简写为 eye（n），即为 n 维单位矩阵 |
| diag（X） | 若 X 是矩阵，则 diag（X）为 X 的主对角线向量，<br>若 X 是向量，diag（X）产生以 X 为主对角线的对角矩阵 |
| tril（A） | 提取一个矩阵的下三角部分 |
| triu（A） | 提取一个矩阵的上三角部分 |
| rand（m，n） | 产生 0 ~ 1 间均匀分布的随机矩阵，m = n 时简写为 rand（n） |
| randn（m，n） | 产生均值为 0，方差为 1 的标准正态分布随机矩阵，m = n 时简写为 randn（n） |

4. 通过后缀为 . m 的命令文件产生。

例如：文件 data. m，其中包括正文：

```
A = [  1  2  3
       4  5  6
       7  8  9]
```

在命令窗口中输入 >> data（回车），则执行 data. m，可以产生名为 A 的矩阵。

## 二、数组运算

数组运算由线性代数的矩阵运算符“ * ”、“/”、“ \ ”、“^”前加一点来表示，即为“. * ”、“. /”、“. \ ”、“. ^”。注意没有“. + ”、“. - ”运算。对于数组运算来说，总认为那

种运算对被运算数组中的每个元素平等地实施同样的操作。其运算是按元素对元素方式进行的。

对于数组的加和减运算与矩阵运算相同，所以“+”、“-”既可被矩阵接受又可被数组接受。数组的乘用符号“.*”表示，如果 A 与 B 矩阵具有相同阶数，则 A.*B 表示 A 和 B 单个元素之间的对应相乘。例如：x=[1 2 3]；y=[4 5 6]。

z = x.*y

z = 4  10  18

数组的左除（.\）与数组的右除（./）以及数组的乘方（.^）的计算方法与乘法类似。

## 三、函数

MATLAB 提供了大量的内部函数，可以通过 help 查询。例如：sqrt（开方）、log（常用对数）、log10（以 10 为底的对数）、sin（正弦函数）等。常见的数学函数如表 1-3-2 所示。

**表 1-3-2　　常见的数学函数**

| 函数名 | 数学计算功能 | 函数名 | 数学计算功能 |
|---|---|---|---|
| abs（x） | 实数的绝对值或复数的幅值 | floor（x） | 对 x 朝 -∞ 方向取整 |
| acos（x） | 反余弦 arccos x | gcd（m，n） | 求正整数 m 和 n 的最大公约数 |
| acosh（x） | 反双曲余弦 arccosh x | imag（x） | 求复数 x 的虚部 |
| angle（x） | 在四象限内求复数 x 的相角 | lcm（m，n） | 求正整数 m 和 n 的最小公倍数 |
| asin（x） | 反正弦 arcsin x | log（x） | 自然对数（以 e 为底数） |
| asinh（x） | 反双曲正弦 arcsinh x | log10（x） | 常用对数（以 10 为底数） |
| atan（x） | 反正切 arctan x | real（x） | 求复数 x 的实部 |
| atan2（x，y） | 在四象限内求反正切 | rem（m，n） | 求正整数 m 和 n 的 m/n 之余数 |
| atanh（x） | 反双曲正切 arctanh x | round（x） | 对 x 四舍五入到最接近的整数 |
| ceil（x） | 对 x 朝 +∞ 方向取整 | sign（x） | 符号函数：求出 x 的符号 |
| conj（x） | 求复数 x 的共轭复数 | sin（x） | 正弦 sin x |
| cos（x） | 余弦 cos x | sinh（x） | 反双曲正弦 sinh x |
| cosh（x） | 双曲余弦 cosh x | sqrt（x） | 求实数 x 的平方根：$\sqrt{x}$ |
| exp（x） | 指数函数 $e^x$ | tan（x） | 正切 tan x |
| fix（x） | 对 x 朝原点方向取整 | tanh（x） | 双曲正切 tanh x |

这些函数都遵循下列规则：对于：

$$A=\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \vdots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}$$

经过函数 f（·）后得：

$$f(A)=\begin{pmatrix} f(a_{11}) & \cdots & f(a_{1n}) \\ \vdots & \vdots & \vdots \\ f(a_{m1}) & \cdots & f(a_{mn}) \end{pmatrix}$$

例：要计算 $\sin\frac{\pi}{6}$，可键入：>> y = sin(pi/6)；

得：

```
y =
   0.5000
```

如果键入：

```
x = [0, pi/6, pi/3, pi/2, 2*pi/3, 5*pi/6, pi];
y = sin(x)
```

得：

```
y =
   0   0.5000   0.8660   1.0000   0.8660   0.5000   0.0000
```

如果键入：

```
x = [0, pi/6; pi/3, pi/2];
y = sin(x)
```

得：

```
y =
   0   0.5000
   0.8660   1.0000
```

其他的函数的用法与此类似。

## 四、矩阵操作

在 MATLAB 中可以对矩阵进行任意操作，包括改变它的形式、选取子矩阵、扩充矩阵、旋转矩阵等。

1. 选取子矩阵。

其中最重要的操作符为“：”，它的作用是取出选定的行与列。

例如：

A（:,:）代表 A 的所有元素；

试比较 A（:），将 A 按列的方向拉成长的 1 列（向量）；

A（:，J）代表 A 的第 J 列；

A（J：K）代表 A（J），A（J+1），…，A（K），如同 A（:）的第 J 到第 K 个元素；

A（:，J：K）代表 A（:，J），A（:，J+1），…，A（:，K），以此类推。

2. 对矩阵可以进行各种各样的旋转、变形、扩充。



MATLAB 中有内部函数 fliplr，它对矩阵进行左右翻转，函数 flipud 对矩阵进行上下翻转。

例：

```
x =
    1  2  3
    4  5  6
fliplr(x)
ans =
    3  2  1
    6  5  4
flipud(x)
ans =
    4  5  6
    1  2  3
```

矩阵的转置用符号“ ' ”表示，如：

```
A = [1  2  3;   4  5  6;   7  8  9]
B = A'
B =
    1  4  7
    2  5  8
    3  6  9
```

符号“ ' ”为矩阵的转置，如果 Z 为复矩阵，则 Z'为它的复数共轭转置，非共轭转置使用 Z.' 或 conj(Z') 求得。

reshape 用于改变矩阵的形状，如：

```
A = [A; [10  11  12]]
A =
    1    2    3
    4    5    6
    7    8    9
    10   11   12
```

则 reshape（A，2，6）

```
ans =
    1   7  2   8  3   9
    4  10  5  11  6  12
```

可见，reshape 是将矩阵元素以列为单位进行重组，原来 4×3 的矩阵变为了 2×6 的矩阵。那么以下的语句也不难理解了，它将矩阵 A 按列打开（size 函数返回矩阵 A 的行数与列数）。

reshape（A，1，size（A，1）* size（A，2）），它等价于 A（:）'。

还有函数 rot90，它可以将矩阵进行各种 90 度的旋转；tril 及 triu 取出矩阵的下三角及上三角部分等。

# 第四章

# 绘　　图

## 一、绘制二维图形

绘图命令 plot，绘制 x－y 坐标图；log 命令绘制对数坐标图；semilogx 和 semilogy 命令绘制半对数坐标图；polor 命令绘制极坐标图。

绘制二维图形的基本命令是 plot（）函数，它的基本调用格式为：

- plot（X，'s'）
- X 是实向量时，以该向量元素的下标为横坐标、元素值为纵坐标画出一条连续曲线。
- X 是实矩阵时，则按列绘制每列元素值相对其下标的曲线。图中曲线数等于 X 阵列数。
- X 是复数矩阵时，则按列分别以元素实部和虚部为横、纵坐标绘制多条曲线。
- s 是用来指定线型、色彩、数据点形的选项字符串。它可以缺省，此时线型、色彩将由 MATLAB 的默认设置确定。
- plot（X，Y，'s'）
- X、Y 是同维向量时，绘制以 X、Y 元素为横、纵坐标的曲线。
- X 是向量，Y 是有一维与 X 等维的矩阵时，则绘制出多根不同色彩的曲线。曲线数等于 Y 阵的另一维数，X 被作为这些曲线的共同横坐标。
- X 是矩阵，Y 是向量时，情况与上相同，只是曲线都以 Y 为共同纵坐标。
- X、Y 是同维矩阵时，则以 X、Y 对应列元素为横、纵坐标分别绘制曲线，曲线条数等于矩阵的列数。
- s 的意义与其在 plot（X，'s'）格式中的意义相同。
- plot（X1，Y1，'s1'，X2，Y2，'s2'，…）
- 在此格式用于在同一图形窗口中绘制多条曲线。每个绘线“三元组”（X，Y，'s'）的结构和作用，与 plot（X，Y，'s'）相同。不同的“三元组”之间没有约束关系。

## 二、曲线的线型、颜色和数据点的形状

在 MATLAB 中可对绘制的曲线的线型、颜色和数据点的形状进行设定。其表示方法如



表 1－4－1 所示。

**表 1－4－1　　　　曲线的线型、颜色和数据点的标记**

| 线型 | 点标记 | 颜色 |
|---|---|---|
| －　实线 | .　点 | y　黄 |
| :　虚线 | o　小圆圈 | m　棕色 |
| －.　点划线 | ×　叉字符 | c　青色 |
| －－　双划线 | ＋　十字符 | r　红色 |
| | *　星号 | g　绿色 |
| | s　方格 | b　蓝色 |
| | d　菱形 | w　白色 |
| | ^　朝上三角 | k　黑色 |
| | v　朝下三角 | |
| | >　朝右三角 | |
| | <　朝左三角 | |
| | p　五角星 | |
| | h　六角星 | |

## 三、坐标控制指令

坐标控制指令 axis 的用途很多，表 1－4－2 列出其常用功能。

**表 1－4－2　　　　坐标控制指令**

| 坐标轴控制方式、取向和范围 | | 坐标轴的高宽比 | |
|---|---|---|---|
| 指　令 | 含　义 | 指　令 | 含　义 |
| axis auto | 使用缺省设置 | axis equal | 纵、横轴采用等长刻度 |
| axis manual | 使当前坐标范围不变 | axis fill | 在 manual 方式下起作用，使坐标充满整个绘图区 |
| axis off | 取消轴背景 | axis image | 纵、横轴采用等长刻度，且坐标框紧贴数据范围 |
| axis on | 使用轴背景 | axis normal | 缺省矩形坐标系 |
| axis ij | 矩阵式坐标，原点在左上方 | axis square | 产生正方形坐标系 |
| axis xy | 普通直角坐标，原点在左下方 | axis tight | 把数据范围直接设为坐标范围 |
| axis (V)<br>V = [x1, x2, y1, y2];<br>V = [x1, x2, y1, y2, z1, z2]; | 人工设定坐标范围。设定值：二维，4 个；三维，6 个 | axis vis3d | 保持高宽比不变，用于三维旋转时避免图形大小变化 |

注：坐标范围设定向量 V 中的元素必须服从：x1 < x2，y1 < y2，z1 < z2。V 的元素值允许取 inf 或 － inf，那意味着上限或下限是自动产生的，即坐标范围半自动确定

## 四、网格线、坐标框以及多层叠绘指令

grid：是否画分格线的双向切换指令。
grid on：画出分格线。
grid off：不画分格线。
box：坐标形式在封闭式和开启式之间切换指令。
box on：使当前坐标呈封闭形式。
box off：使当前坐标呈开启形式。
hold on：使当前轴及图形保持而不被刷新，准备接受此后绘制的新曲线。
hold off：使当前轴及图形不再具备不刷新的性质。
hold：当前图形是否具备刷新性质的双向切换开关。

## 五、图形的加注命令

title（S）：标注图名。
xlabel（S）：标注横坐标轴名。
ylabel（S）：标注纵坐标轴名。
legend（S1，S2，…）：添加绘制曲线所用线型、色彩或数据点形图例。
text（xt，yt，S）：在图面（xt，yt）坐标处添加字符注释。

## 六、绘制子图

在绘图过程中，经常要把几个图形在同一个图形窗口中在不同的坐标系中表现出来，而不是简单地叠加。这就用到函数 subplot。其调用格式如下：

subplot（m，n，p）；

subplot 函数把一个图形窗口分割成 m×n 个子区域，用户可以通过参数 p 调用各子绘图区域进行操作。子绘图区域的编号为按行从左至右编号。

例如：把 sin x，cos x，atan x，sin xcos y 画在一个窗口内，可键入：

```
x=0: pi/6: 2*pi; y=x;
z1=sin(x); z2=cos(x); z3=atan(x);
subplot(2,2,1); plot(x,z1,'r',x,z2,'g')
subplot(2,2,2); plot(x,z3,'m')
subplot(2,2,3); [x,y]=meshgrid(x,y); z4=sin(x).*cos(y);
mesh(x,y,z4); subplot(2,2,4); surfc(x,y,z4)
```

# 七、绘制三维图形

1. 空间曲线的绘制

绘制空间曲线的基本命令为：

plot3（x，y，z）;

plot3（x，y，z，'s'）;

plot3（x1，y1，z1，'s1'，x2，y2，z2，'s2'，…）;

其中，x，y，z 是同维的向量或矩阵。当它们是矩阵时，以它们的列对应元素为空间曲线上点的坐标。s 是对线型、颜色的设定，这一点与二维曲线时的情形相同。

2. 曲面的绘制

绘制函数 z = f（x，y）所代表的三维空间曲面，需要做以下数据准备：

（1）确定自变量 x，y 的取值范围和取值间隔。x = x1：dx：x2；y = y1：dy：y2；

（2）构造 x - y 平面上的自变量“网格”矩阵。[X，Y] = meshgrid（x，y）;

（3）计算在自变量采样“网格点”上的函数值，即 Z = f（X，Y）。

绘制空间曲面网格图的命令为 mesh（x，y，z）。

如果 x、y 是向量，则要求 x 的长度 = 矩阵 z 的列维；y 的长度 = 矩阵 z 的行维。以 $z_{ij}$ 为纵坐标，x 的第 i 个分量为横坐标，y 的第 j 个分量为纵坐标绘网格图。

如果是同维矩阵，则数据点的坐标分别取自这三个矩阵。其他常用的曲面图绘制命令有：

meshc（x，y，z）：带等高线的网格图。

waterfall（x，y，z）：瀑布水线图。

surf（x，y，z，'c'）：可着色的曲面图。

surfc（x，y，z）：带等高线的可着色的曲面图。

以上这些命令都可用来绘制曲面图，用法与 mesh 完全一样。

例如：要想画马鞍面，可输入如下命令：

```
x = -3：1/16：3；y = x；
[x，y] = meshgrid（x，y）;        %（生成绘图时所需的 x - y 坐标）
z = -x.^2 + y.^2；
mesh（x，y，z）
```

# 第五章

# 关系运算和逻辑运算

## 一、关系运算（见表1－5－1）

**表1－5－1**　　　　**关系运算符**

| < | 小于 |
|---|---|
| < = | 小于等于 |
| > | 大于 |
| > = | 大于等于 |
| = = | 等于 |
| ~ = | 不等于 |

运算法则：

如果两个比较量 a、b 是标量，那么，当 a、b 之间的关系成立时输出值为 1；否则输出值为 0。

如果两个比较量 a、b 是相同维数的数组，那么就按标量的运算法则，对 a、b 的对应元素进行运算，最后的输出结果为一个与 a（或 b）同维的 0－1 数组。

如果 a 是标量，b 是数组，那么按标量的运算法则将 a 与 b 的每个元素逐一比较，最后的输出结果为一个与 b 同维的 0－1 数组。

在算术运算和关系运算中，算术运算优先。

## 二、逻辑运算（见表1－5－2）

**表1－5－2**　　　　**逻辑运算符**

| & | 与 |
|---|---|
| \| | 或 |
| ~ | 非 |
| Xor（x，y） | 异或 |

运算法则：

参与逻辑运算的量称为逻辑量，非零逻辑量为“真”，用1表示；零逻辑量为“假”，用0表示。

如果参与逻辑运算的两个量a、b都是标量，那么：

| | |
|---|---|
| a & b | 当a与b全为非零时，运算结果为“1”；否则为“0” |
| a \| b | a与b中只要有一个非零，运算结果为“1” |
| ~a | 当a是零时，运算结果为“1”；否则为“0” |
| xor（x，y） | 异或运算。x或y非零（真）返回1，x和y都是零（假）或都是非零（真）返回0 |

如果参与逻辑运算的两个量a、b是相同维数的数组，那么就按标量的运算法则，对a、b的对应元素进行运算，最后的输出结果为一个与a（或b）同维的0-1数组。

如果参与逻辑运算的a是标量，b是数组，那么就按标量的运算法则，将a与b的每个元素进行运算，最后的输出结果为一个与b同维的0-1数组。

逻辑“非”是一个一元运算符，也服从数组运算规则。

在算术、关系、逻辑运算中，算术运算的最优先，其次是关系运算，再其次是逻辑运算。

## 三、关系与逻辑函数

除上面关系与逻辑操作符外，MATLAB还有如下的关系与逻辑函数：

any（x）：如果在一个向量x中，任何元素是非零，返回1；矩阵x中的每一列有非零元素，返回1。

all（x）：如果在一个向量x中，所有元素非零，返回1；矩阵x中的每一列所有元素非零，返回1。

下面是一些测试特殊值或条件存在的函数，返回逻辑值：

isfinite：元素有限，返回真值。

isempty：参量为空，返回真值。

isglobal：参量是一个全局变量，返回真值。

ishold：当前绘图保持状态是'ON'，返回真值。

isieee：计算机执行IEEE算术运算，返回真值。

isinf：元素无穷大，返回真值。

isletter：元素为字母，返回真值。

isnan：元素为不定值，返回真值。

isreal：参量无虚部，返回真值。

isspace：元素为空格字符，返回真值。

isstr：参量为一个字符串，返回真值。

# 第六章

# MATLAB 程序设计

## 一、控制流语句

MATLAB 也有控制流语句，用于控制程序的流程。跟其他程序语言一样，MATLAB 也有顺序、选择和循环三种基本程序结构。主要语句有 if 语句、switch 语句、for 循环语句、while 循环语句等。

1. 选择结构

选择结构是根据给定条件的成立与否，分别执行不同的语句。MATLAB 用于实现选择结构的语句有 if 语句和 switch 语句。

（1）if 条件语句。

在许多情况下，命令的序列必须根据关系的判断而有条件地执行。在编程语言里，这种逻辑由 if 语句来提供。按其复杂程度分为：

① 单分支结构：

```
if 表达式
    执行语句
end
```

若在表达式中的所有元素为真（非零），则执行 if 和 end 语言之间的执行语句。

② 双分支结构：

```
if 表达式 1
    执行语句 1
else
    执行语句 2
end
```

在这里，如果表达式为真，则执行第一组命令；如果表达式是假，则执行第二组命令。

③ 多分支结构：

```
if 表达式 1
    执行语句 1
elseif 表达式 2
```

```
    执行语句 2
......
elseif 表达式 n
    执行语句 n
else
    执行语句 n+1
end
```

最后的这种形式，如果表达式 1 为真，则执行语句 1；接下来的关系表达式不检验，跳过其余的 if - else - end 结构。而且，最后的 else 命令可有可无。

例：对函数 $f(x)=\begin{cases} x & x<1 \\ 2x-1 & 1\leqslant x<10 \\ 3x-11 & x\geqslant 10 \end{cases}$

可编如下的程序：

```
if  x<1
y=x;
elseif  x>=1&x<10
y=2*x-1;
else
y=3*x-11;
end
```

（2） switch 语句。

根据表达式的不同取值，分别执行不同的语句。switch 语句的一般表达形式为：

```
switch 表达式
    case 表达式 1
        执行语句 1
    case 表达式 2
        执行语句 2
......
    case 表达式 n
        执行语句 n
    otherwise
        执行语句 n+1
end
```

MATLAB 首先计算表达式的值，然后将它依次与各个 case 指令后的检测值进行比较，当比较结果为真时，就执行相应的语句组，然后跳出 switch 结构。如果所有的比较结果都为假，则执行 otherwise 后面的语句组，然后跳出 switch 结构。最后的 otherwise 指令可以不出现。

2. 循环结构

循环结构是按照给定的条件，重复执行指定的语句。MATLAB 用于实现循环结构的语

句有 for 语句和 while 语句。

（1） for 循环语句。

for 循环语句的一般表达形式为：

```
for     循环变量 = 表达式
        循环体语句
end
```

其中表达式是一个矩阵，因为 MATLAB 中都是矩阵，矩阵的列被一个接一个地赋值给循环变量，然后运行循环体语句。

通常表达式是一些 m: n 或 m: k: n 仅有一行的矩阵，并且它的列是个简单的标量。但如果表达式为矩阵，则循环变量为向量。

例：求 $S=1+2+3+\cdots+50$，可编程如下：

```
s=0;
for k=1:50
s=s+k;
end
```

（2） while 循环语句。

while 循环语句用来控制一个或一组语句在某逻辑条件下，重复预先确定或不确定的次数。while 循环语句的一般表达形式为：

```
while     表达式
          循环体语句
end
```

当表达式的所有运算为非零值时，循环体语句组将被执行。如果判断条件是向量或矩阵的话，可能需要 all 或 any 函数作为判断条件。

例：对于上面同样的问题，可编程如下：

```
S=0; k=0;
while k<51
S=S+k; k=k+1;       %  当条件 k<51 时，反复执行语句 S=S+k，k=k+1
end
```

以上这个例子事先已知循环次数是 51，下面再看一个预先不能确定循环次数的例子。

例：用迭代法求 $x_{k+1}=e^{-x_k}$　$k=0, 1, 2, \cdots$

求解方程 $x-e^{-x}=0$ 的根。初始值 $x_0=0.5$，相对误差限 $\varepsilon=10^{-8}$，编程如下：

```
ep=10^(-8); dx=1; x0=0.5; k=0;
while dx>ep
k=k+1;
x=exp (-x0);
dx=abs (x-x0)/(1+abs (x));
x0=x;
end
```

# 二、创建M文件

创建M文件是MATLAB中的非常重要的内容。事实上，正是由于在MATLAB工具箱中存放着大量的M文件，使得MATLAB在应用起来显得简单、方便，且功能强大。如果用户根据自己的需要，开发出适用于自己的M文件，不仅能使MATLAB更加贴近用户自己，而且能使MATLAB的功能得到扩展。

M文件以.m为扩展名，它有两种形式：命令文件（或脚本文件）和函数文件。当用户要运行的命令较多时，如果直接在命令窗口中逐条输入和运行，有诸多不便。此时可通过编写命令文件来解决这个问题。若要运行这些程序行的命令，只需要在命令窗口输入M文件名即可。另外，MATLAB的许多功能则需要用户通过编写函数文件来实现。它提供了MATLAB的外部函数。用户为解决一个特定问题而编写的大量的外部函数可放在MATLAB工具箱中，这样的一组外部函数形成一个专用的软件包。

1. 命令文件的建立。

进入MATLAB命令窗口后，选择“file”下拉式菜单中的“new”选择“M－File”，或在命令窗口中输入“edit M文件名”，或在点击工具栏中的“新建”按钮。在M文件编辑窗口中，写入符合语法规则的命令。编写完命令文件后，选择“file”下拉式菜单中的“save”项，然后依提示输入一个文件名。至此，完成了命令文件的创建。

2. 函数文件的建立。

函数文件的创立方法与命令文件的创立方法完全一样，只是函数文件的第一句可执行语句是以function引导的定义语句，并且输入文件名时要与定义语句中的函数名相同。

3. 命令文件和函数文件的调用。

建立了命令文件后，只要在命令窗口键入命令文件名，再按回车键，就可执行M文件中所包含的所有命令。

函数文件一旦创建完成，就可像调用其他MATLAB函数一样调用了。例如：

建立一个函数用来计算数组的平均值，将其命名为meanfun.m。

在编辑器中，输入以下命令：

```
function y = meanfun (x)
    % MEAN Average or mean value, For Vectors,
    % MEAN (x) returns the mean value
    % for matrix MEAN (x) is a row vector
    % containing the mean value of each column
     [m, n] = size (x);
    if m = =1
        m = n;
end
y = sum (x)/m;
```

将其保存为meanfun.m，在命令窗口中输入：



```
x = reshape (1:9, 3, 3);
y = meanfun (x)
```

输出结果为：

```
y =
    2   5   8
```

书

# 第七章

# MATLAB 的符号运算

MATLAB 符号运算是通过符号数学工具箱（Symbolic Math Toolbox）来实现的。符号运算工具箱是操作和解决符号表达式的工具箱（函数）集合，主要包括：符号表达式的运算，符号表达式的复合、化简，符号矩阵的运算，符号微积分、符号作图，符号代数方程求解，符号微分方程求解等。另外还有一些用于线性代数的工具，求解逆、行列式、正则型式的精确结果，找出符号矩阵的特征值等。符号运算的结果无由数值计算引入的误差。工具箱还支持可变精度运算，即支持符号计算并能以指定的精度返回结果。

## 一、符号对象的建立

在进行符号运算时，必须先定义基本的符号对象，可以是符号常量、符号变量、符号表达式等。MATLAB 提供了两个建立符号对象的函数：sym 和 syms，两个函数的用法不同。

1. sym 函数。

sym 函数用来建立单个符号量，一般调用格式为：

符号量名 = sym（'符号字符串'）

该函数可以建立一个符号量，符号字符串可以是常量、变量、函数或表达式。

应用 sym 函数还可以定义符号常量，使用符号常量进行代数运算时和数值常量进行的运算不同。如：

```
a = sym（'a'）
b = sym（1/5）
c = sym（'［1/3，cd；c，d］'）
```

2. syms 函数。

函数 sym 一次只能定义一个符号变量，使用不方便。MATLAB 提供了另一个函数 syms，一次可以定义多个符号变量。syms 函数的一般调用格式为：

syms　符号变量名 1　符号变量名 2　…　符号变量名 n

用这种格式定义符号变量时不要在变量名上加字符串分界符（'），变量间用空格而不要用逗号分隔。如：

```
syms a b c
```



## 二、符号变量确定原则

1. 除了 i 和 j 之外，字母位置最接近 x 的字母；若距离相等，则取 ASCII 码大的视为默认的符号变量；

2. 若没有除了 i 与 j 以外的字母，则视 x 为默认的符号变量；

3. 可利用函数 findsym（string，N）来确定众多符号中的第 N 个符号变量。例如：键入 findsym（3 * a * b + y^2，1），即可得到答案 y。更多的例子见表 1 -7 -1：

**表 1 -7 -1**

| 符号表达式 | 默认符号变量 |
|---|---|
| a * x^2 + b * x + c | x |
| 1/(4 + cos（t）) | t |
| 4 * x/y | x |
| 2 * a + b | b |
| 2 * i | x |

## 三、符号表达式的运算

1. 符号表达式的化简。

MATLAB 提供的对符号表达式化简的函数有：

simplify（s）：应用函数规则对 s 进行化简。

simple（s）：调用 MATLAB 的其他函数对表达式进行综合化简，并显示化简过程。

2. 符号表达式与数值表达式之间的转换。

利用函数 sym 可以将数值表达式变换成它的符号表达式。

函数 numeric 或 eval 可以将符号表达式变换成数值表达式。

3. 符号表达式的替换。

用给定的数据替换符号表达式中的指定的符号变量。

subs（f，x，a）

用 a 替换字符函数 f 中的字符变量 x，a 是可以是数/数值变量/表达式或字符变量/表达式。

## 四、符号微积分

下面着重介绍一些与微积分有关的指令，这些指令都需要符号表达式作为输入变量。

1. 求和 $\sum_{k} S(k)$。

symsum（S）：对通项 S 求和，其中 k 为变量，且从 0 变到 k－1。

symsum（S，v）：对通项 S 求和，其中 v 为变量，且 v 从 0 变到 v－1。

symsum（S，a，b）：对通项 S 求和，其中 k 为变量，且从 a 变到 b。

symsum（S，v，a，b）：对通项 S 求和，其中 v 为变量，且 v 从 a 变到 b。

例：求 $\sum_{i=0}^{k-1} i$，键入

```
k = sym ('k')        % k 是一个符号变量;
symsum (k)
```

得 ans = 1/2 * k^2 - 1/2 * k

例：求 $\sum_{k=0}^{10} k^2$，键入：

```
symsum (k^2, 0, 10)
```

得 ans = 385

例：求 $\sum_{k=0}^{+\infty} \frac{x^k}{k!}$ 键入

```
symsum ('x'^k/sym ('k! '), k, 0, inf),
```

得 ans = exp（x）

这最后的一个例子是无穷项求和。

2. 求极限。

limit（P）：表达式 P 中自变量趋于零时的极限。

limit（P，a）：表达式 P 中自变量趋于 a 时的极限。

limit（P，x，a，'left'）：表达式 P 中自变量 x 趋于 a 时的左极限。

limit（P，x，a，'right'）：表达式 P 中自变量 x 趋于 a 时的右极限。

例：求 $\lim_{x \to 0} \frac{\sin x}{x}$，键入

```
P = sym ('sin(x)/x');
limit (P)
```

得 ans = 1

例：求 $\lim_{x \to 0^+} \frac{1}{x}$ 键入

```
P = sym ('1/x');
limit (P, 'x', 0, 'right')
```

得 ans = inf

例：求 $\lim_{h \to 0} \frac{\sin(x+h) - \sin x}{h}$，键入：

```
P = sym ('(sin(x+h) - sin(x))/h'); h = sym ('h');
limit (P, h, 0)
```

得 ans = cos（x）

例：求 $(\lim_{x\to-\infty}\left(1+\frac{a}{x}\right)^x, \lim_{x\to-\infty} e^{-x})$，键入

```
v = sym ('[(1 + a/x)^x, exp (-x)] ');
limit (v, 'x', inf, 'left')
```

得 ans = [exp(a), 0]

3. 求导数。

diff (S, v)：求表达式 S 对变量 v 的一阶导数。

diff (S, v, n)：求表达式 S 对变量 v 的 n 阶导数。

例如：设 $A=\begin{pmatrix}\frac{1}{1+a} & \frac{b+x}{\cos x}\\ 1 & e^{x^2}\end{pmatrix}$，求 $\frac{dA}{dx}$键入命令：

```
syms a b x;
A = [1/(1 + a), (b + x)/cos(x); 1, exp(x^2)];
diff(A, 'x')
```

得 ans = [0, 1/cos(x) + (b + x)/cos(x)^2 * sin(x)]
[0, 2 * x * exp(x^2)]

例：求 $y=\sin x+e^x$ 的三阶导数，键入命令：

```
diff('sin(x) + x * exp(x) ', 3)
```

得 ans = - cos(x) + 3 * exp(x) + x * exp(x)

例：设 $A=\begin{pmatrix}x\sin y & x^n+y\\ \frac{1}{xy} & e^{xyi}\end{pmatrix}$，求 A 的先对 x 再对 y 的混合偏导数。可键入命令：

```
S = sym ('[x * sin(y), x^n + y; 1/x/y, exp(i * x * y)] ');
dsdxdy = diff (diff(S, 'x'), 'y')
```

得：dsdxdy = [ cos(y), 0]
[ 1/x^2/y^2, i * exp(i * x * y) - y * x * exp(i * x * y)]

例：求 $y=(\ln x)^x$ 的导数。可键入命令：

```
p = ' (log (x))^x';
p1 = diff (p, 'x')
```

得：p1 = log (x)^x * (log (log (x)) + 1/log (x))

例：求 $y=x\cdot f(x^2)$ 的导数。可键入命令：

```
p = 'x * f (x^2) '; p1 = diff (p, 'x')
```

得：p1 = f(x^2) + 2 * x^2 * D(f)(x^2)

例：求由方程 $xy=e^{x+y}$确定的隐出数的导数。可键入命令：

```
p = 'x * y(x) - exp (x + y (x)) '; p1 = diff (p, 'x')
```

得：p1 = y(x) + x * diff(y(x), x) - (1 + diff(y(x), x)) * exp (x + y (x))

再键入

```
p2 = 'y + x * dy - (1 + dy) * exp (x + y) = 0';
dy = solve (p2, 'dy')%
```

把 dy 作为变量解方程

得 dy = - (y - exp (x + y))/(x - exp (x + y))

4. 求 Taylor 展开式。

taylor(f, v)　f 对 v 的五阶麦克劳林展开。

taylor(f, v, n)　f 对 v 的 n - 1 阶麦克劳林展开。

taylor(f, v, n, a)　f 对 v 在 v = a 处的 n - 1 阶泰勒展开式。

例：求 $\sin x \cdot e^{-x}$ 的 7 阶麦克劳林展开。可键入

f = sym('sin(x) * exp( - x)');F = taylor(f, 8)得

F = x - x^2 + 1/3 * x^3 - 1/30 * x^5 + 1/90 * x^6 - 1/630 * x^7

例：求 $\sin x \cdot e^{-x}$ 在 x = 1 处的 7 阶 Taylor 展开。可键入

```
f = sym('sin(x) * exp( - x)'); F = taylor(f, 8, 1) 得
F = sin(1) * exp( - 1) + ( - sin(1) * exp( - 1) + cos(1) * exp( - 1)) * (x - 1)
    - cos(1) * exp( - 1) * (x - 1)^2
    + (1/3 * sin(1) * exp( - 1) + 1/3 * cos(1) * exp( - 1)) * (x - 1)^3
    - 1/6 * sin(1) * exp( - 1) * (x - 1)^4
    + (1/30 * sin(1) * exp( - 1) - 1/30 * cos(1) * exp( - 1)) * (x - 1)^5
    + 1/90 * cos(1) * exp( - 1) * (x - 1)^6
    + ( - 1/630 * cos(1) * exp( - 1) - 1/630 * sin(1) * exp( - 1)) * (x - 1)^7
```

5. 求积分。

int (P)：对表达式 P 进行不定积分。

int (P, v)：以 v 为积分变量对 P 进行不定积分。

int (P, v, a, b)：以 v 为积分变量，以 a 为下限，b 为上限对 P 进行定积分。

例：求 $\int \frac{-2x}{(1+x^2)^2}dx$，可键入

int ('-2 * x/(1 + x^2)^2')

得　ans = 1/(1 + x^2)

例：求 $\int \frac{x}{(1+z^2)}dz$，可键入

键入 int ('x/(1 + z^2)', 'z')

得 ans = atan (z) * x

例：求 $\int_0^1 x\ln(1+x)dx$，可键入

int ('x * log (1 + x)', 0, 1) 得

ans = 1/4

例：求 $\int_{\sin t}^{\ln t} 2x dx$ 可键入：

int ('2 * x', 'sin (t)', 'log (t) ') 得：

ans = log (t)^2 - sin (t)^2

6. 求符号方程的解。

(1) 线性方程组的求解。

线性方程组的形式为 A * X = B；其中 A 至少行满秩。

X = linsolve（A，B）

输出方程的特解 X。

例：解方程组 $\begin{pmatrix} \cos t & \sin t \\ \sin t & \cos t \end{pmatrix} X = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$。键入

```
A = sym ('[cos (t), sin (t); sin (t), cos (t)]'); B = sym ('[1; 1]');
c = linsolve (A, B)
c = [ 1/(sin (t) + cos (t))]
    [ 1/(sin (t) + cos (t))]
```

（2）代数方程的求解。

solve（P，v）

对方程 P 中的指定变量 v 求解。v 可省略。

solve（P1，P2，…，Pn，v1，v2，…，vn）

对方程 P1，P2，…，Pn 中的指定变量 v1，v2，…，vn 求解。

例：解 $p + \sin x = r$，可输入

solve（'p + sin (x) = r'）得：

ans = - asin (p - r)

例：解 $\begin{cases} x^2 + xy + y = 3 \\ x^2 - 4x + 3 = 0 \end{cases}$，可输入：

```
P1 = 'x^2 + x * y + y = 3'; P2 = 'x^2 - 4 * x + 3 = 0';
[x, y] = solve (P1, P2) 得：
x = [1]
    [3]
y = [   1]
    [ -3/2]
```

解 $\begin{cases} a + u^2 + v^2 = 0 \\ u - v = 1 \end{cases}$，可输入：

```
P1 = 'a + u^2 + v^2 = 0'; P2 = 'u - v = 1'; [u, v] = solve (P1, P2, 'u', 'v') 得：
u = [ 1/2 + 1/2 * ( -1 -2 * a)^(1/2)]
    [ 1/2 - 1/2 * ( -1 -2 * a)^(1/2)]
v = [ -1/2 + 1/2 * ( -1 -2 * a)^(1/2)]
    [ -1/2 - 1/2 * ( -1 -2 * a)^(1/2)]
```

对于有些无法求出解析解的非线性方程组，MATLAB 只给出一个数值解。这一点可以从表示解的数字不被方括号括住而确定。

例：解 $\begin{cases} \sin(x + y) - ye^x = 0 \\ x^2 - y = 2 \end{cases}$，键入：

```
[x, y] = solve ('sin (x + y) - exp (x) * y = 0', 'x^2 - y = 2') 得：
x = -6.0173272500593065641097297117905
y = 34.208227234306296508646214438330
```

由于这两个数字没有被 [ ] 括住，所以它们是数值解。另外，可利用 solve 来解线性方程组的通解。

例：解$\begin{pmatrix}2 & 7 & 3 & 1\\3 & 5 & 2 & 2\\9 & 4 & 1 & 7\end{pmatrix}X=\begin{pmatrix}6\\4\\2\end{pmatrix}$键入

P1 ='2 * x1 +7 * x2 +3 * x3 + x4 =6'; P2 ='3 * x1 +5 * x2 +2 * x3 +2 * x4 =4';

P3 ='9 * x1 +4 * x2 + x3 +7 * x4 =2';

u = solve (P1, P2, P3, 'x1', 'x2', 'x3', 'x4')

Warning: 3 equations in 4 variables。

u = x1: [1x1 sym]

x2: [1x1 sym]

x3: [1x1 sym]

x4: [1x1 sym]

可以看到：屏幕提示"有 3 个方程 4 个变量"，意为解不唯一（有时会提示解不唯一）。且输出的是解的结构形式。为进一步得到解，可输入：

u. x1, u. x2, u. x3, u. x4, 得：

ans = x1

ans = -5 * x1 -4 * x4

ans = 11 * x1 +9 * x4 +2

ans = x4

这样就得到了原方程组的通解。

7. 解符号微分方程。

解符号微分方程的命令格式为：dsolve ('eq1', 'eq2', …)。

其中 eq 表示相互独立的常微分方程、初始条件或指定的自变量。默认的自变量为 t。如果输入的初始条件少于方程的个数，则在输出结果中出现常数 c1，c2 等字符。关于微分方程的表达式有如下的约定：字母 y 表示函数，Dy 表示 y 对 t 的一阶导数；Dny 表示 y 对 t 的 n 阶导数。

例如：求$\begin{cases}\dfrac{dx}{dt}=y\\\dfrac{dy}{dt}=-x\end{cases}$

的解可键入：[x, y] =dsolve ('Dx = y', 'Dy = -x') 得

x = cos (t) * C1 + sin (t) * C2

y = -sin (t) * C1 + cos (t) * C2

dsolve 中的输入变量最多只能有 12 个，但这并不妨碍解具有多个方程的方程组，因为可以把多个方程或初始条件定义为一个符号变量进行输入。

例如：求 $\dfrac{df}{dt}=3f+4g$，$\dfrac{dg}{dt}=-4f+3g$，f (0) =0，g (0) =1 的解。可输入指令：

P ='Df =3 * f +4 * g, Dg = -4 * f +3 * g'; v ='f (0) =0, g (0) =1';

[f, g] =dsolve (P, v)



f = exp (3 * t) * sin (4 * t)

g = exp (3 * t) * cos (4 * t)

注意：微分方程表达式中字母 D 必须大写。

例如：求解微分方程$\begin{cases}\dfrac{d^3y}{dx^3}=-y\\ y(0)=1,\ y'(0)=0,\ y''(0)=0\end{cases}$

可输入

y = dsolve ('D3y = -y', 'y (0) = 1, Dy (0) = 0, D2y (0) = 0', 'x') 得：

y = (1/3 + 2/3 * exp (1/2 * x) * cos (1/2 * 3^(1/2) * x) * exp (x))/exp (x)

最后看一个解非线性微分方程的例子：

dsolve ('(Dy)^2 + y^2 = 1', 'y (0) = 0', 'x')

ans = [ sin (x)]
      [ -sin (x)]

对于无法求出解析解的非线性微分方程，屏幕将提示出错信息。





# 第二部分

## MATLAB 软件实验

## 实验一

# MATLAB 的基本操作

（验证性实验）

### 一、实验目的

1. 熟悉 MATLAB 的界面和基本操作。
2. 掌握 MATLAB 的基本运算方法。
3. 掌握 MATLAB 中帮助命令的使用方法。

### 二、实验原理

1. 打开 MATLAB，进入 MATLAB 界面后，在 Desktop 菜单下可对各窗口的打开及隐藏进行操作。



2. 在 Current Directory 命令框中设定当前工作目录。在此后的工作中，生成的 M 文件和数据文件都保存在此目录中。

3. 对命令窗口（Command Window）的常用操作包括 clf，clc，clear，edit，exit，cd，md 等。

4. 数组的创建。

（1）一维数组的创建。

- 直接输入法，例如：

x = ［2 pi/2 sqrt（2）3 +5i］

- 冒号生成法，例如：

x = pi：pi/10：pi

- 定数线性采样法。

该法在设定的“总点数”下，均匀采样生成一维“行”数组。

格式如下：

x = linspace（a，b，n）

（2）矩阵的创建。

- 直接输入法，对于较小数组，从键盘上直接输入最简便。二维数组必须有以下 3 个要素：

➢ 整个输入数组必须以方括号“［ ］”为其首尾；

➢ 数组的行与行之间必须用“；”或回车键隔离；

➢ 每一行内的元素之间必须用逗号“，”或空格隔开；

- 利用 M 文件创建和保存数组。

对于今后经常需要调用的数组，尤其是比较大而复杂的数组，为它专门建立一个 M 文件是值得的。如创建和保存数组 AM 的 MyMatrix. m 文件。

```
AM =[101，102，103，104，105，106，107，108，109；…
     201，202，203，204，205，206，207，208，209；…
     301，302，303，304，305，306，307，308，309]；
```

保存此文件，并且文件名为：MyMatrix. m

- 利用特殊矩阵函数生成法。

系统中提供的几个用于构造特殊矩阵的命令，见表 2 - 1 - 1。

**表 2 - 1 - 1**

| 函　数 | 功　能 | 函　数 | 功　能 |
|---|---|---|---|
| compan | 伴随阵 | vander | Vandermonde 矩阵 |
| diag | 对角阵 | zeros | 元素全为 0 的矩阵 |
| magic | 魔方矩阵 | ones | 元素全为 1 的矩阵 |
| eye | 对角线上元素为 1 的矩阵 | rand | 元素服从均匀分布的随机矩阵 |
| meshgrid | 由两个向量生成的网格矩阵 | randn | 元素服从正态分布的随机矩阵 |

5. 数组的寻访。

（1）一维数组的寻访。

x (n)：提取 x 的第 n 个元素。

x ( [a b])：提取 x 的第 a、b 个元素。

(2) 二维数组的寻访。

A (r, c)：由 A 的第 r 行和第 c 列上的元素组成。

A (r, :)：由 A 的第 r 行和全部列上的元素组成。

A ( [a b], c)：由 A 的第 a、b 行和第 c 列上的元素构成。

A (:, c)：由 A 的全部行和第 c 列上的元素组成。

A (:)："单下标全元素寻访"。

A (s)："单下标寻访"，单下标的索引的第 s 个元素。

6. 矩阵运算与数组运算。

矩阵运算：MATLAB 运算的基本单元是由实数或复数元素构成的矩阵。标量和向量是特殊的矩阵，标量为 $1\times1$ 阶矩阵，而向量是只有一行或一列的矩阵。从矩阵角度看，MATLAB 中的运算和命令趋于自然表达形式。

数组运算：数组是指由一组实数或复数排成的长方阵列（Array）。数组运算是指无论在数组上施加什么运算（加、减、乘、除或函数），总认为那种运算对被运算数组中的每个元素平等地实施同样的操作。注意，进行数组乘、除和幂运算时要在运算符前加点，例如：.*，./，.^。

7. 系统的在线帮助。

(1) help 命令。

- 当不知系统有何帮助内容时，可直接输入 help 以寻求帮助：

>> help（回车）

- 当想了解某一主题的内容时，例如输入：

>> help elfun

- 当想了解某一具体的函数或命令的帮助信息时，例如输入：

>> help rem

(2) lookfor 命令。

现需要完成某一具体操作，不知有何命令或函数可以完成，例如输入：

>> lookfor line（查找与直线、线性问题有关的函数）

(3) 利用系统的 Demos 菜单来查看实现某种功能的程序示例或工具箱的使用方法。

## 三、实验内容

1. 启动 MATLAB，熟悉其各个窗口和菜单。
2. MATLAB 实现简单算术运算。
3. 矩阵的输入。
4. 子数组的寻访和赋值。
5. 用矩阵函数生成矩阵。
6. 学习使用 MATLAB 的帮助功能。





# 四、实验报告

## MATLAB 的基本操作

实验名称：____________________ 实验日期：______年____月____日

姓名：____________________ 班级学号：____________________

成绩：____________________

### （一）实验内容及步骤

1. 进入 MATLAB 工作环境，熟悉各窗口的功能。

（1）双击桌面图标■，或从“开始”菜单打开 MATLAB. exe，启动 MATLAB。

（2）查看 MATLAB 界面各窗口的布局、了解其功能，并完成各窗口之间的切换。

（3）设置当前工作目录。在 D 盘创建 mymatlab 目录，并将其作为当前工作目录。今后的实验过程中以此目录作为当前工作目录。

2. 计算 $5.54^2 + \cos\left(\frac{3}{5}\pi\right)\sqrt{17.89} \div 3.5 - e^2$ 的值。

（1）在命令窗口（Command Window）中输入程序：

____________________________________________________

（2）按回车键运行，如果出现“Error”（出错信息），则应找出原因并改正，再运行。

（3）运行结果：__________________________________。

3. 输入矩阵 $A=\begin{bmatrix}4 & 2 & 8\\ 2 & 5 & 6\\ 3 & 1 & 9\end{bmatrix}$，$B=\begin{bmatrix}1 & 1 & 1\\ 2 & 2 & 2\\ 3 & 3 & 3\end{bmatrix}$，在命令窗口中执行下列表达式，掌握其含义：

A (2, 3)　A (:, 2)　A (3,:)　A (:, 1: 2: 3)　A (:, 3). *B (:, 2)　A (:, 3) *B (2,:)

A * B　A. * B　A^2　A. ^2　B/A　B. /A

（1）在命令窗口键入____________________________，生成矩阵 A；键入____________________________，生成矩阵 B。

（2）记录执行以下表达式的结果：

A (2, 3) =______，A (6) =______，A (:, 2) =______，A (3,:) =______，



A（:，1：2：3）= ______，A（:，3）．* B（:，2）= ______，A（:，3）* B（2,:）= ______，A * B = ______，A．* B = ______，A^2 = ______，A．^2 = ______，B/A = ______，B．/A = ______，B．\ A = ______。

4. 产生一个 5 阶魔方矩阵，将矩阵的第 3 行 4 列元素赋值给变量 a；将由矩阵第 2，3，4 行第 2，5 列构成的子矩阵赋值给变量 b。

（1）在命令窗口键入 >> clear 和 >> clc，观察命令窗口内容的变化。

clear 命令的作用是______________________________。

clc 命令的作用是______________________________。

（2）利用 magic（）函数产生魔方矩阵，并将此矩阵赋给变量 M。

程序______________________________。

（3）将矩阵 M 的第 3 行第 4 列元素赋值给变量 a。

程序______________________________。

（4）将由矩阵 M 第 2，3，4 行第 2，5 列构成的子矩阵赋值给变量 b。

程序______________________________。

5. 建立如下矩阵。

（1）$\begin{pmatrix} 2010 & 0 & \cdots & 0 \\ 0 & 2010 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 2010 \end{pmatrix}_{10\times 10}$

程序：______________________________。

（2）$\begin{pmatrix} 0 & 6 & \cdots & 6 \\ 6 & 0 & \cdots & 6 \\ \vdots & \vdots & & \vdots \\ 6 & 6 & \cdots & 0 \end{pmatrix}_{8\times 8}$

程序：______________________________。

6. 设有分块矩阵 $A = \begin{bmatrix} E_{3\times 3} & R_{3\times 2} \\ O_{2\times 3} & D_{2\times 2} \end{bmatrix}$，其中 E，R，O，D 分别为单位阵、随机阵、零阵和对角阵 $\begin{bmatrix} e & 0 \\ 0 & 2 \end{bmatrix}$。

（1）生成单位阵 E，程序：______________________________。

（2）生成随机阵 R，程序：______________________________。

（3）生成零阵 O，程序：______________________________。

（4）生成对角阵 D，程序：______________________________。

（5）生成矩阵 A，程序：______________________________。

7. 查看 plot 函数的帮助文档，并查看 Demos 中关于 plot 的示例程序。

（1）在命令窗口输入 >> help plot，查看 plot 函数的帮助文档。

（2）点击“help”菜单，选择“Demos”子菜单，在“MATLAB”中选择“Graphics”，再选择“2 - D Plots”，即可看到关于 plot 函数的示例程序。



## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 在 MATLAB 运算中，A * B 与 A. * B 以及 B./A 与 B.\A 之间有什么区别?

2. 实验小结。





实验二

# MATLAB 的绘图功能

## （验证性实验）

## 一、实验目的

1. 了解 MATLAB 的图形窗口及其基本操作。
2. 掌握 MATLAB 绘制二维平面图形的命令。
3. 掌握 MATLAB 绘制三维立体图形的命令。
4. 了解一些常用绘图命令及绘图标注。

## 二、实验原理

1. 绘制图形的一般步骤，见表 2 -2 -1。

**表 2 -2 -1**

| 步　　骤 | | 典型指令 |
|---|---|---|
| 1 | 曲线数据准备<br>先取一个参变量采样向量<br>然后计算各坐标数据向量 | t = pi * (0：100)/100<br>x = f1 (t)；y = f2 (t)；z = f3 (t) |
| | 三维曲面数据<br>产生自变量采样向量<br>产生自变量“格点”矩阵<br>计算自变量“格点”矩阵相应的函数值矩阵 | x = x1：dx：x2<br>y = y1：dy：y2<br>[X, Y] = meshgrid (x, y)<br>Z = f (X, Y) |
| 2 | 选定图形窗及子图位置：<br>缺省时，打开 Figure No. 01，或当前窗，当前子图<br>可用指令指定图形窗号和子图号 | figure (1)　% 指定 1 号图形窗<br>subplot (2, 2, 3)　% 指定三号子图 |
| 3 | 调用二维、三维曲线绘图指令：<br>线型、色彩、数据点形 | plot (x, y, 'r: ')　% 用红虚点画二维线<br>plot3 (x, y, z, 'b - ')　% 用蓝色实线画曲线 |
| | 调用三维曲面绘图指令 | mesh (X, Y, Z) |

续表

| 步　骤 | | 典型指令 |
|---|---|---|
| 4 | 设置轴的范围、坐标分格线 | axis（[x1, x2, y1, y2]）　%平面坐标范围<br>axis（[x1, x2, y1, y2, z1, z2]）　%三维坐标范围<br>grid on　%坐标分格线 |
| 5 | 图形注释：<br>图名、坐标名、图例、文字说明 | tiitle ('调制波形')　%图名<br>xlabel ('t'); ylabel ('y')　%轴名<br>legend ('sin (t) ', 'sin (t) sin (9t) ')　%图例<br>text (2, 0. 5, 'y=sin (t) sin (9t) ')　%文字说明 |
| 6 | 着色、明暗、灯光、材质处理 (仅对三维图形使用) | colomap, shading, light, material |
| 7 | 视点、三度 (横、纵、高) 比 (仅对三维图形使用) | view, aspect |
| 8 | 图形的精细操作 (图柄操作)：利用对象属性值设置; 利用图形窗工具条进行 | get, set |

2. 常用的曲线绘图命令。

（1）二维作图函数：plot，其基本调用形式为：

plot（x，y，s）

图2－2－1中，以x作为横坐标，y作为纵坐标.s是图形显示属性的设置选项。例如：

```
x = -pi: pi/10: pi;
y = cos (x);
plot (x, y, '-. gd', 'linewidth', 4, 'markeredgecolor', 'r', 'markerfacecolor', 'c',)
```

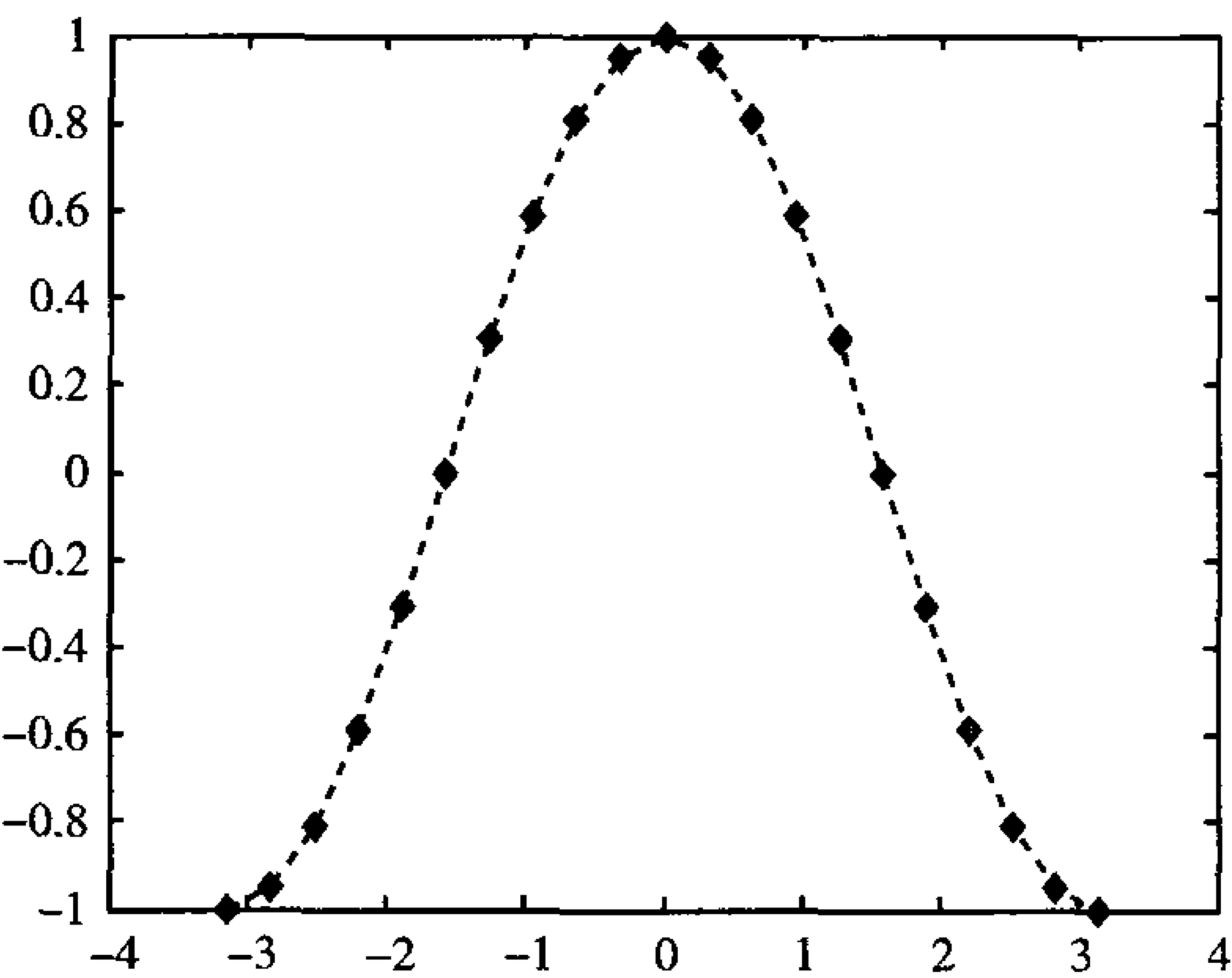



**图2－2－1**

绘图时，可以制定标记的颜色和大小，也可以用图形属性制定其他线条特征，这些属性包括：

linewidth：指定线条的粗细。

markeredgecolor：指定标记的边缘色。

markerfacecolor：指定标记表面的颜色。

markersize：指定标记的大小。

若在一个坐标系中画多条曲线，则 plot 的调用格式如下：

plot（x1，y1，s1，x2，y2，s2，…）

在使用函数 plot 时，应当注意到当两个输入量同为向量时，向量 x 与 y 必须维数相同，而且必须同是行向量或者同是列向量。

（2）隐函数绘图。

MATLAB 提供了一个 ezplot 函数绘制隐函数图形，对于隐函数 $f=f(x, y)$，ezplot 函数的调用格式为：

ezplot（f）：在默认区间 $-2\pi<x<2\pi$ 和 $-2\pi<y<2\pi$ 绘制 $f(x, y)=0$ 的图形。

ezplot（f，[xmin，xmax，ymin，ymax]）：在区间 xmin < x < xmax 和 ymin < y < ymax 绘制 $f(x, y)=0$ 的图形。

ezplot（f，[a，b]）：在区间 $a<x<b$ 和 $a<y<b$ 绘制 $f(x, y)=0$ 的图形。

ezplot（x，y，[tmin，tmax]）：在区间 tmin < t < tmax 绘制 $x=x(t)$ 和 $y=y(t)$ 的图形。

（3）三维曲线作图函数：plot3，其调用格式与 plot 函数类似（见图 2-2-2）：

plot3（x，y，z，s）. 例如：

```
t=(0：0.02：2)*pi；
x=sin（t）；y=cos（t）；z=cos（2*t）；
plot3(x，y，z，'b-'，x，y，z，'bd')，view([-82，58])，box on，legend('链'，'宝石')
```

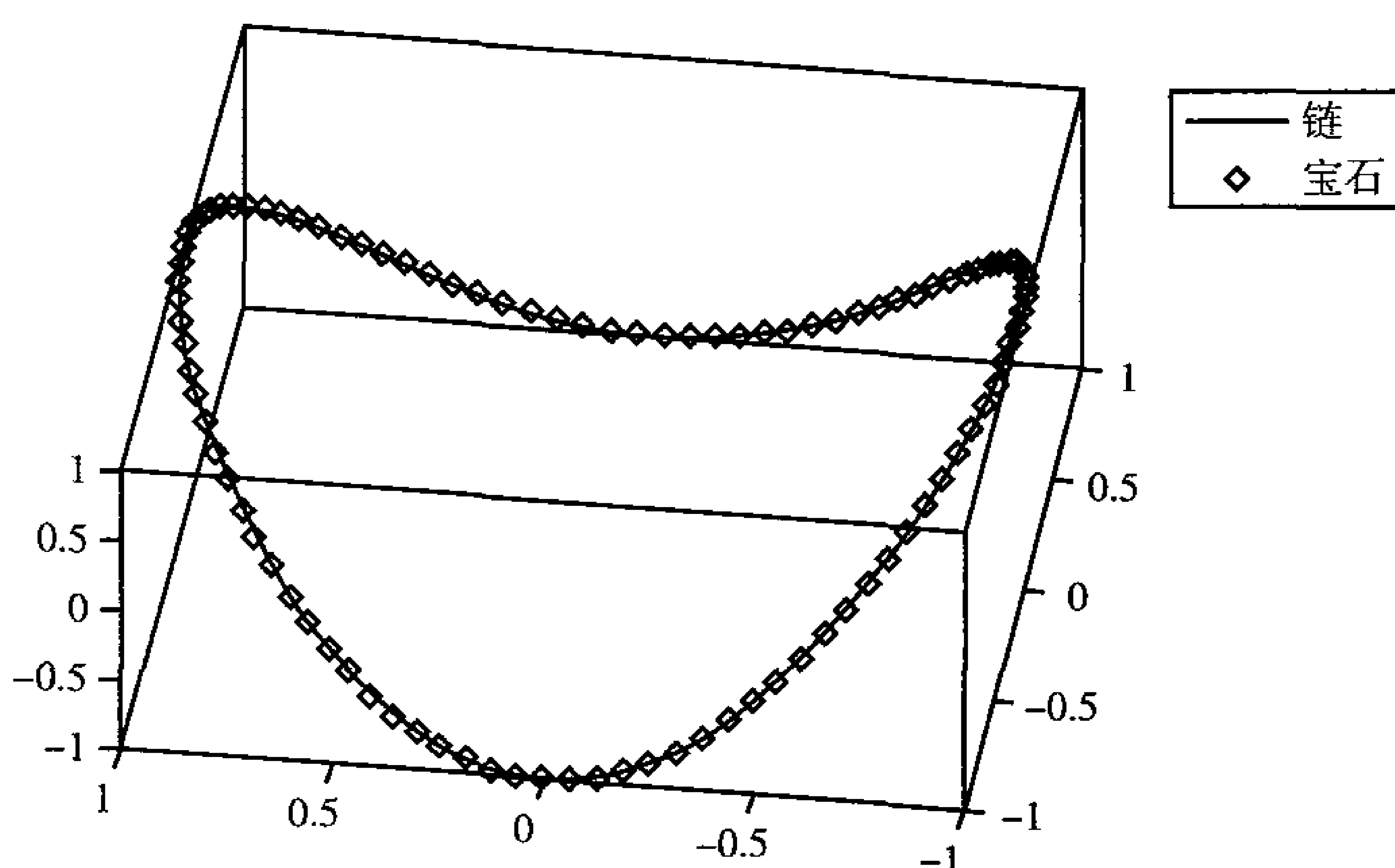



**图 2-2-2**

3. 空间曲面作图函数。

（1）网格图。

利用在 x-y 平面的矩形网格点上的 z 轴坐标值，MATLAB 定义了一个网格曲面。MATLAB 通过将邻接的点用直线连接起来形成网状曲面，其结果好像在数据点有结点的渔网，调用格式为：

mesh（z），mesh（x，y，z）和 mesh（x，y，z，c）。

其中，mesh（x，y，z，c）画出颜色由 c 指定的三维网格图。若 x、y 均为向量，则 length（x）=n，length（y）=m，[m，n] =size（z）。

例如，用 MATLAB 的函数 peaks 可以画一个简单的曲面。

[X，Y，Z] =peaks（30）；

mesh（X，Y，Z）

grid，xlabel（'x - axis'），ylabel（'y - axis '），zlabel（' z - axis'）

title（' MESH of PEAKS '）

输出见图 2 - 2 - 3：

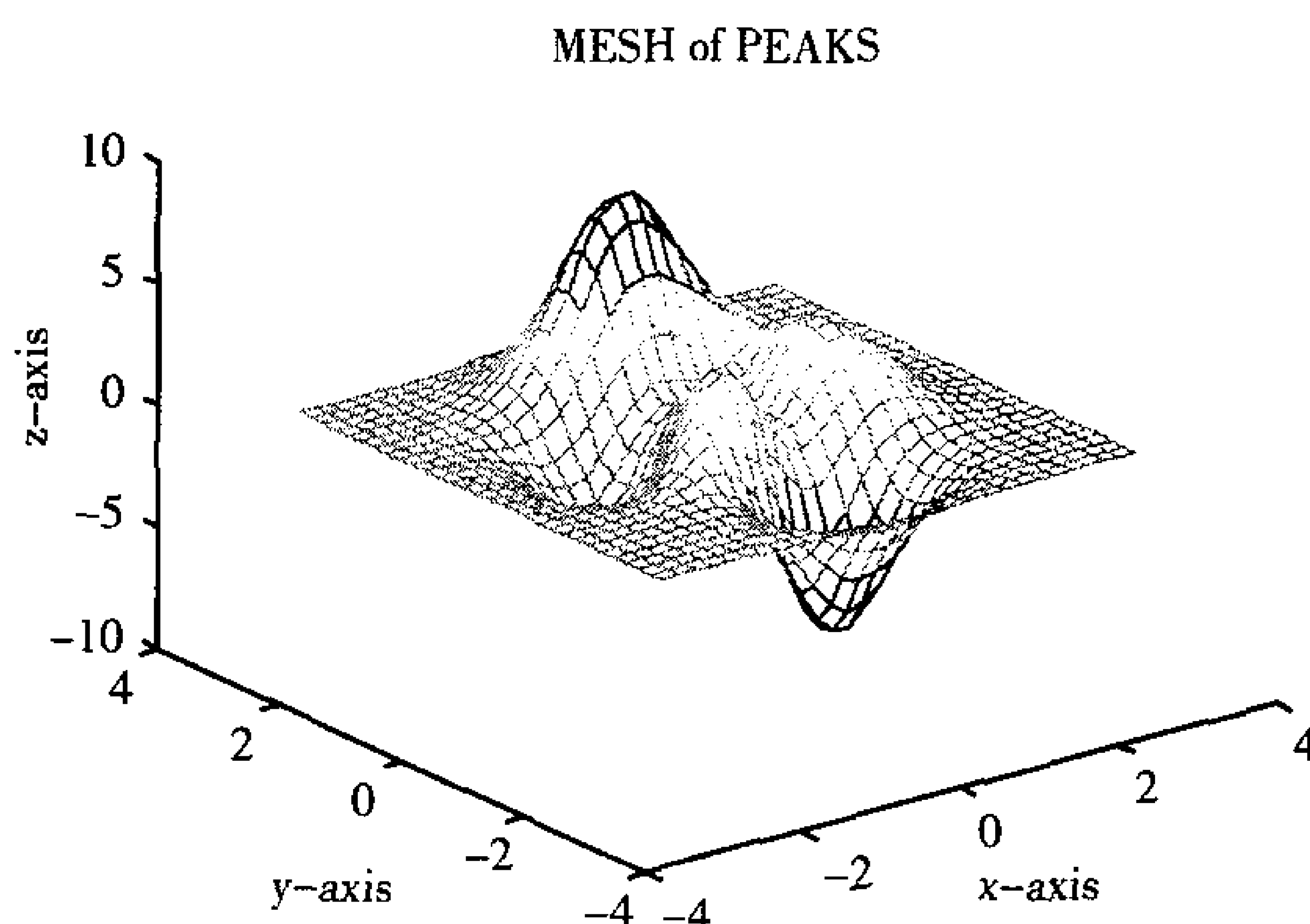



**图 2 - 2 - 3**

（2）曲面图。

曲面图除了各线条之间的空当（称作补片）用颜色填充以外，和网格图看起来是一样的。这种图一般使用函数 surf 来绘制。调用格式与 mesh 类似。例如：

[X，Y，Z] = sphere（30）；　　%　sphere 为画球体命令

surf（X，Y，Z）

grid，xlabel（'x - axis '），ylabel（'y - axis '），zlabel（'z - axis '）

title（'SURF of PEAKS'）

图形见图 2 - 2 - 4：

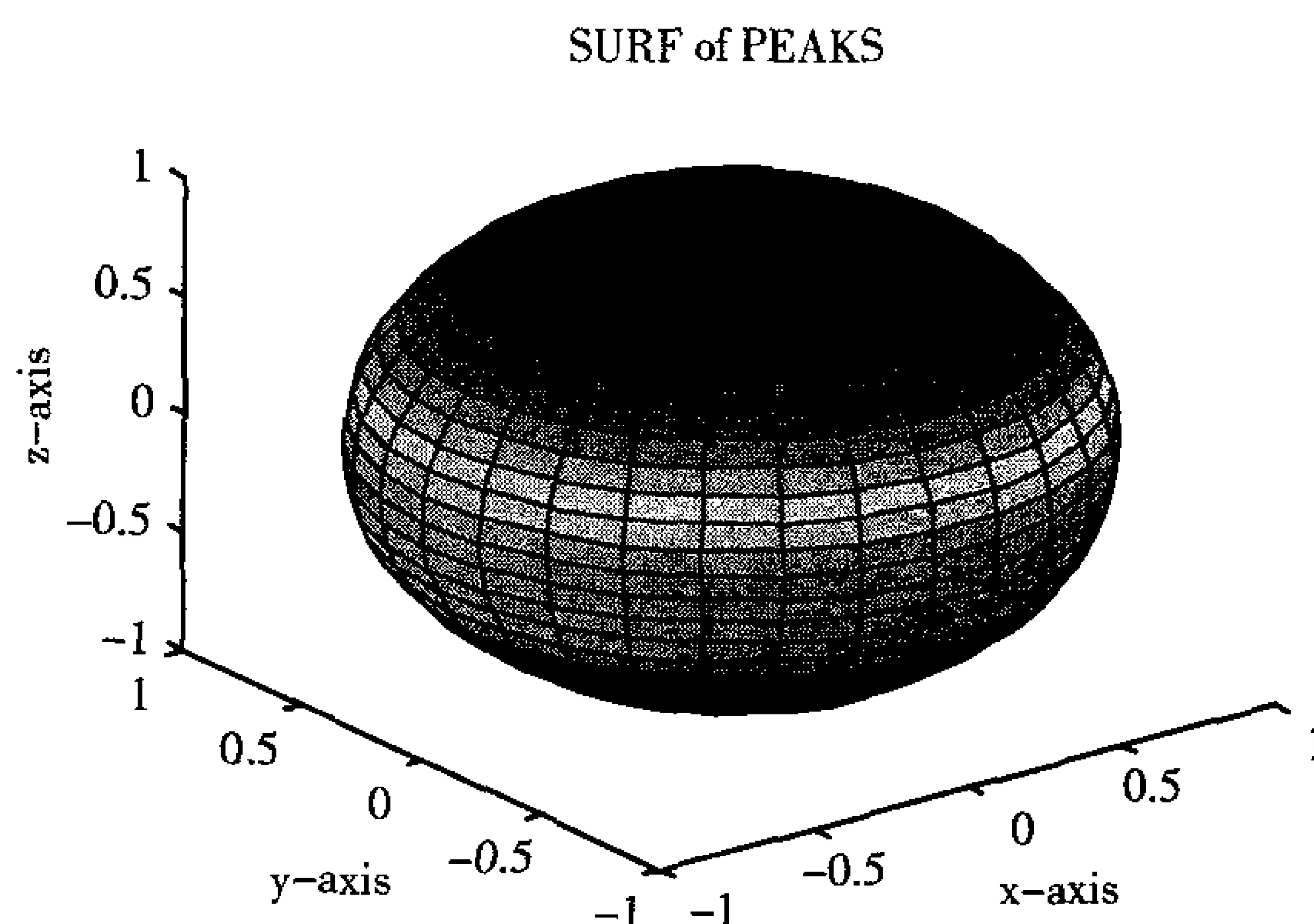



**图 2 - 2 - 4**

（3）ezmesh 用于参数方程确定的三维曲面网格图。调用格式为：

ezmesh（x，y，z）

其中 x = x（s，t），y = y（s，t），z = z（s，t）。画图区域默认为：-2 * pi < s < 2 * pi

且 -2 * pi < t < 2 * pi。或者用格式：ezmesh（x，y，z，[smin，smax，tmin，tmax]）

（4）ezsurf 用于参数方程确定的三维曲面图。调用格式与 ezmesh 类似。

4. 网格数据点生成函数 meshgrid，调用格式。

[x，y] =meshgrid（m，n），

这里的 m，n 为给定的向量，可以定义网格划分区域和划分方法。矩阵 x 和矩阵 y 是网格划分后的数据矩阵。

5. 图形的透视。

MATLAB 在采用缺省设置画 mesh 图形时，对叠压在后面的图形采取了消隐措施，但有时却需要透视效果。为此，MATLAB 提供了一个控制消隐的指令如下：

hidden off：透视被叠压的图形

hidden on：消隐被叠压的图形

6. 图形的镂空。

利用"非数" NaN，对图形进行镂空处理。下面是为了得到圆域上的旋转抛物面而将圆域以外的部分镂空处理的程序（见图 2-2-5）。

```
x = -2：0.05：2；
y = -2：0.05：2；
[X，Y] =meshgrid（x，y）；
Z =X.^2 + Y.^2；
i =find（X.^2 + Y.^2 >4）；
Z（i）=nan；        %镂空
surf（X，Y，Z）；
```

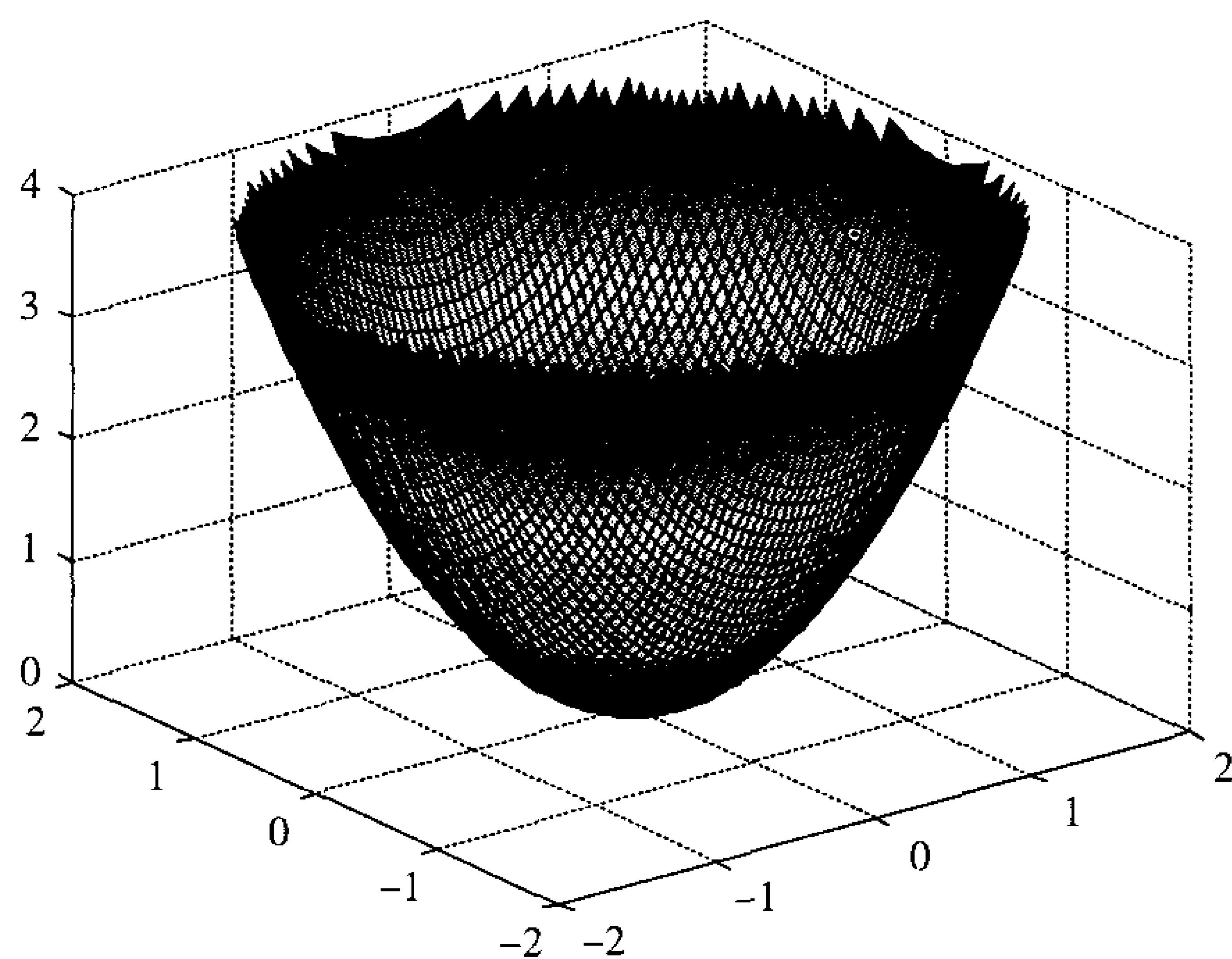



图 2-2-5

其他图形命令：

bar（x，y，选项）：条形图。

stairs（x，y，选项）：阶梯图。

stem（x，y，选项）：火柴杆图。



fill（x1，y1，选项1，x2，y2，选项2，…）：填充图。

7. 图形的修饰命令。

（1）视点处理命令。

MATLAB 提供了设置视点的函数 view，其调用格式为：view（az，el）

其中 az 为方位角，el 为仰角，它们均以度为单位。系统缺省的视点定义为方位角 $-37.5°$，仰角 $30°$。

（2）色彩处理命令。

可用 colormap 函数设置绘图颜色。其调用格式为：colormap（[r g b]）

其中 r，g，b 都是 0－1 之间的数。或者用格式：colormap（s）

s 为颜色映像。下面举几个常用的例子见表 2－2－2。

**表 2－2－2**

| 颜色映像 | 相应的颜色系 | 颜色映像 | 相应的颜色系 |
|---|---|---|---|
| autumn | 红黄色系 | hsv | 色调饱和色系 |
| gray | 线性灰色系 | hot | 黑红黄白色系 |
| cool | 青和洋红色系 | pink | 柔和色系 |

（3）三维表面图形的着色。

三维表面图实际上就是在网格图的每一个网格片上涂上颜色。surf 函数用缺省的着色方式对网格片着色。除此之外，还可以用 shading 命令来改变着色方式。

shading faceted：命令将每个网格片用其高度对应的颜色进行着色，但网格线仍保留着，其颜色是黑色。这是系统的缺省着色方式。

shading flat：命令将每个网格片用同一个颜色进行着色，且网格线也用相应的颜色，从而使得图形表面显得更加光滑。

shading interp：命令在网格片内采用颜色插值处理，得出的表面图显得最光滑。

## 三、实验内容

1. MATLAB 的图形窗口的打开与关闭。
2. MATLAB 二维图形的绘制。
3. 在一个图形窗口绘制多个图形。
4. 图形线条颜色、线条类型的设置。
5. 图形标注、坐标控制、图形修饰处理。
6. 子图函数的使用。
7. MATLAB 三维图形的绘制。



# 四、实验报告

## MATLAB 的绘图功能

实验名称：______________　　　实验日期：______年____月____日
姓名：______________　　　班级学号：______________
成绩：______________

### （一）实验内容及步骤

1. MATLAB 的图形窗口的打开与关闭。

（1）在“File”菜单下选择“New”，在“New”子菜单下选择“Figure”，或在命令窗口中输入 >> Figure（回车）即可新建一个新的图形窗口。

（2）观察图形窗口的各个菜单后，关闭图形窗口。

2. 运行该段程序，掌握 plot 命令，并注释程序。

```
v = ['-1'; '+1'];
t = [0: 0.05: 2*pi];    %______________________________。
plot (t, sin (t));    %
xlabel ('time (0 - -2\ pi) ');    %______________________________。
ylabel ('value');    %______________________________。
text (pi/2, 0.9, ['\ uparrowsin (\ pi*1/2) =', v (2,:)]);
text (pi*3/2, -0.9, ['\ downarrowsin (\ pi*3/2) =', v (1,:)]);
text (0, -0.6, ['Date: ', date]);    %______________________________。
```

在上面程序中任意改动某些参数，观察图形的变化（这里可改动 plot 中的可选参数，如线型、颜色等，还可将文本的标注位置挪动等等）。

3. 运行该段程序，掌握 subplot 和 pause 命令，并注释程序。

```
clf    %______________________________。
t=0: 0.1: 4*pi;
y=exp (-0.1*t).*sin (t);
figure (1)    %______________________________。
subplot (2, 2, 1),    %______________________________。
stem (t, y)    %______________________________。
title ('stem (t, y) '),    %______________________________。
```



```
pause      %____________________________________________。
subplot (2, 2, 2),      %_______________________________________。
stairs (t, y)      %____________________________________________。
title ('stair (t, y) '),
pause
subplot (2, 2, 3),      %_______________________________________。
bar (t, y)
title ('bar (t, y) '),
pause
subplot (2, 2, 4),
fill ( [0, t, 4*pi], [0, y, 0], 'r')      %_______________________________。
title ('fill (t, y," r") ')
```

4. 利用函数 plot 在一个坐标系中画以下几个函数图像，要求采用不同颜色、不同线型、不同的符号标记。

$x=\sin(t)$，$t\in(0, 2\pi)$，数据点用星号表示，线型为黑色虚线。

$y=\cos(t)$，$t\in(0, 2\pi)$，数据点用方块表示，线型为红色实线。

$z=\sin(2t)$，$t\in(0, 2\pi)$，数据点用小圆圈表示，线型为蓝色点线。

（1）在命令窗口（Command Window）中输入程序：

（2）按回车键运行，如果出现“Error”（出错信息），则应找出原因并改正，再运行。

（3）查看运行结果。

（4）如果要将所得图形插入到 word 文档中，应如何操作。

______________________________________________________________

5. 绘制三维螺旋线：$x=2(\cos t+t\sin t)$，$y=2(\sin t-t\cos t)$，$z=1.5t$，$t\in[0, 10\pi]$。要求给相应的坐标轴和标题附加标注，螺线为蓝色点线，线条粗细设置为3。

（1）生成三维螺旋线的程序。

（2）给 x、y、z 轴分别加标注“x”、“y”、“z”，并给图形加上标题“三维螺旋线”，并将线条粗细设置为3。

程序：

运行结果。

6. 已知曲面方程 $z=\frac{\cos x\sin y}{y}$，在 $x\in[-1.5\pi,\ 1.5\pi]$，$y\in[-1.5\pi,\ 1.5\pi]$ 的取值范围内，用建立子窗口的方法在同一个图形窗口中绘制出三维线图、网格图、光滑曲面图以及镂空图。镂空图要求将 $z<-0.1$ 部分镂空，在每张子图上加标题标注。

（1）生成绘制三维曲线图和曲面图的数据点。

程序：

（2）设置子图，在前三张子图上分别绘制三维曲线图，网格图和光滑曲面图。

程序：

（3）绘制镂空图。将 $z<-0.1$ 的数据点的 z 坐标取值设置为“nan”。

程序：

（4）运行结果。

7. 作出图 2-2-6 所示的三维图形（选做）。

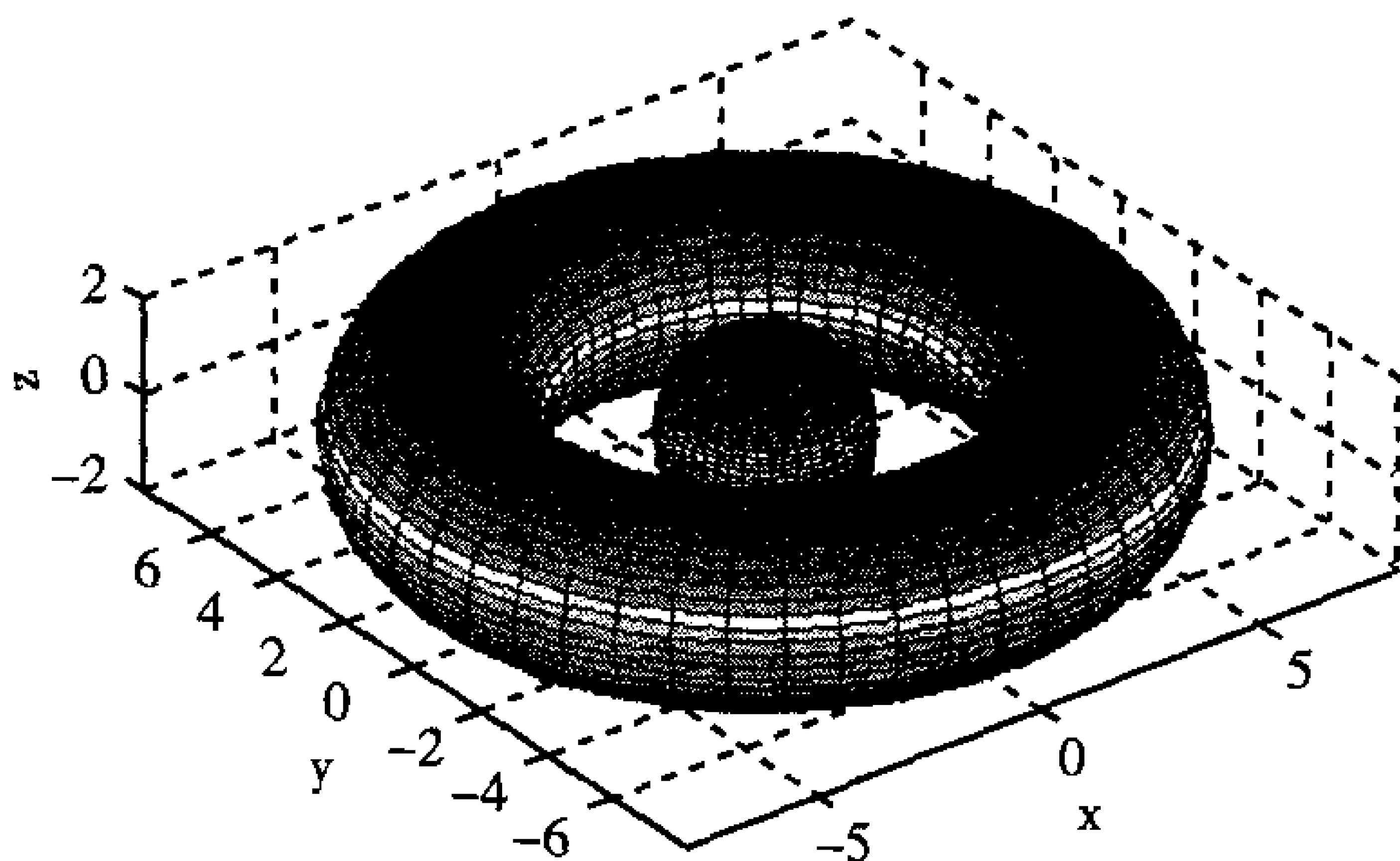



**图 2-2-6**

提示：图形为圆环面和球面的组合，圆环面的方程为：



$(\sqrt{x^2+y^2}-R)^2+z^2=r^2$，$R=6$，$r=2$，而圆环面的参数方程为：

$$\begin{cases} x=(R+r\cos u)\cos v, \\ y=(R+r\cos u)\sin v, \quad u\in[0,2\pi],\ v\in[0,2\pi] \\ z=r\sin u, \end{cases}$$

程序：

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 在画三维曲面图的时候，可以省略 meshgrid 语句吗？为什么？画三维曲线图需要使用 meshgrid 函数吗？

2. 实验内容的第 6 题中，y 的取值为 0 的时候会出现什么现象？如何避免此现象的出现？

3. 实验小结。

实验三

# MATLAB 的程序设计
（设计性实验）

## 一、实验目的

1. 掌握函数 M 文件与命令 M 文件的建立与编辑。
2. 掌握 MATLAB 编程语言的基本控制结构与控制转移语句。
3. 掌握 MATLAB 程序的调试。

## 二、实验原理

1. 输入输出函数

（1）数据的输入。input 函数，其调用格式为：

x = input（提示信息）

其中，提示信息为字符串应用单引号括起来，该命令要求用户输入 x 的值（可以是数或字符串）。

（2）数据的输出。disp 函数或 fprintf 函数。

disp（X）用于输出变量 X 的值，可以是数值矩阵或字符串。

fprintf 函数用于按指定的格式将变量的值输出到指定的文件。其调用格式为：

fprintf（fid，format，variables）

其中 fid 为文件句柄；若缺省，则将变量的值输出到屏幕上；format 用来指定数据输出时采用的格式，常见的格式有：

%e（采用科学计算形式）。

%f（采用浮点数形式）。

%g（由系统自动选取上述两种格式之一）。

%s（输出字符串）。

format 中还可以使用一些特殊格式，如：

\ n（换行）、\ t（制表符）、\ b（退格）、\\ （反斜杆）、%%（百分号）。

2. 程序控制命令

（1）程序的暂停。

pause（n）

其中 n 是延迟时间，以秒为单位；若缺省，则将暂停程序，直到用户按任意键后继续。

（2）break 和 continue。

break 语句用于终止循环的执行，即跳出最内层循环；continue 语句用于结束本次循环，进行下一次循环。break 和 continue 一般与 if 语句配合使用。

（3）return。

return 语句用于退出正在运行的脚本或函数，通常用在函数文件中。

3. 控制流

（1）if-else-end 结构。

在许多情况下，命令的序列必须根据关系的判断而有条件地执行。在编程语言里，这种逻辑由 if-else-end 结构来提供。最简单的是单分支结构，例如：

```
if rand (1, 1) >0.5,
disp ('Given random number is greater than 0.5.');
end
```

当有两种选择时，可用双分支结构。如：

```
x = input ('x =');
if x > =0
y =1;
else
y = -1;
end
```

当有多种选择时，可使用多分支结构。如：

```
x = input ('x =');
if x >0
y =1;
elseif x <0
y = -1;
else
y =0
end
```

（2）for 循环。

for 循环允许一组命令以固定的和预定的次数重复。for 循环的一般形式是：

```
for 循环变量 = 数组
    循环体
end
```

在 for 和 end 语句之间的循环体，按数组中的每一列执行一次。在每一次迭代中，循环



变量被指定为数组的下一列。例如，

```
for n =1：10
    x（n）=sin（n * pi/10）;
end
```

（3）While 循环

for 循环适用于已知循环次数的情况，而 While 循环适用于循环次数未知的情况。其一般形式是：

```
while 表达式
    循环体
end
```

只要在表达式里的所有元素为真，就执行 while 和 end 语句之间的循环体。通常，表达式的值是一个标量值，但数组值也同样有效。在数组情况下，所得到数组的所有元素必须都为真。例如：

```
num =0; EPS =1;
while（1 + EPS） >1
    EPS = EPS/2;
    num = num + 1;
end
```

4. M 文件、命令文件及函数文件。

（1）M 文件。

用 MATLAB 语言编写的程序，称为 M 文件。这些文件名以 . m 形式结尾，例如文件名为 bessel. m，提供 bessel 函数语句。一个 M 文件包含一系列的 MATLAB 语句，一个 M 文件可以循环地调用它自己。

M 文件有两种类型：

第一类型的 M 文件称为命令文件，它是一系列命令、语句的简单组合。

第二类型的 M 文件称为函数文件，它提供了 MATLAB 的外部函数。用户为解决一个特定问题而编写的大量的外部函数可放在 MATLAB 工具箱中，这样的一组外部函数形成一个专用的软件包。

这两种形式的 M 文件，无论是命令文件，还是函数文件，都是普通的 ASCII 文本文件，可选择编辑或字处理文件来建立。

（2）命令文件。

当一个命令文件被调用时，直接在命令窗口输入其文件名，再按回车键即可。对于进行分析解决问题及做设计中所需的一长串繁杂的命令和解释是很有用的。

例如：一个自编的命令文件 fib. m，用于计算 Fibonnaci 数列。

```
f= [1, 1 ]; i=1;
while f（i）+f（i+1） <1 000
    f（i+2）=f（i）+f（i+1）;
    i=i+1;
end
```

```
disp (f)
```

在命令窗口中键入 fib 命令，按回车执行，将计算出所有小于 1 000 的 Fibonnaci 数列。

要注意的是：文件执行后，f 和 i 变量仍然留在工作空间。

(3) 函数文件。

如果 M 文件的第一行包含 function，这个文件就是函数文件，它与命令文件不同，所定义变量和运算都在文件内部，而不在工作空间。函数被调用完毕后，所定义变量和运算将全部释放。函数文件对扩展 MATLAB 函数非常有用。

例如：一个自编的函数文件 fibf. m，用于求小于 n 的 Fibonnaci 数列。

```
function f = fibf (n)
f = [1, 1]; i = 1;
while f (i) + f (i + 1) < n
    f (i + 2) = f (i) + f (i + 1);
    i = i + 1;
end
disp (f)
```

该函数可与 MATLAB 函数一样使用，例如：

输入：

```
y = fibf (10)
y =
    1    1    2    3    5    8
```

5. 程序的调试。

一般来说，应用程序的错误有两类，一类是语法错误，另一类是运行时的错误。语法错误包括：词法或文法的错误，例如函数名的拼写错、表达式书写错等；程序运行时的错误是指程序的运行结果有错误，这类错误也称为程序逻辑错误。

一般检测逻辑错误的方法：

删除表达式最尾端的分号。

若调试一个单独函数，可在其第一行的函数申明行加上%，并定义输入变量的值，直接以脚本的方式来执行此 M 文件，并保留所有变量于 MATLAB 工作空间中，以便后续查看及调试。

在程序中加入 keyboard 命令，可使程序执行至此处即暂停，并显示“k >”的提示号。此时可任意查看或改变工作空间的任何变量。若要继续执行程序，可在命令窗输入“return”并按下 Enter 键。

使用 MATLAB 程序调试器。MATLAB 的 debug 菜单下为我们提供了调试程序的调试器。首先，可在需要程序停止的命令行上设置断点，断点的设置与取消可在 debug 菜单的 set/clear Breakpoint 选项中设定。设定好断点后，当程序运行至此命令行时，将会临时停止运行，此时，可以查看和更改当前变量的值来调试程序。



## 三、实验内容

1. 命令文件的建立及调用。
2. 函数文件的建立及调用。
3. 三种控制结构。
4. MATLAB 程序的调试。





# 四、实验报告

## MATLAB 程序设计

实验名称：______________　　实验日期：______年____月____日
姓名：______________　　班级学号：______________
成绩：______________

### （一）实验内容及步骤

1. M 文件的建立与打开。

（1）新建一个 M 文件。

- 菜单操作（File→New→M-File）
- 命令操作（edit M 文件名）
- 命令按钮（快捷键）

（2）打开已有的 M 文件。

- 菜单操作（File→Open）
- 命令操作（edit M 文件名）
- 命令按钮（快捷键）
- 双击 M 文件

2. 命令文件的建立与调用。

（1）新建一个 M 命令文件，实现将华氏温度转化为摄氏温度的功能。输入如下程序：

```
clear;
f = input ('Please input Fahrenheit temperature: ');
c = 5 * (f - 32)/9;
fprintf ('The Centigrade temperature is %g \ n', c);
```

（2）将此 M 文件命名为 f2c. m。

（3）调用 f2c. m 文件，计算 f = 100 时的结果。在命令窗口中输入：>> f2c（回车）出现提示输入时，输入 100，运行结果：______________________。

3. 函数文件的建立与调用。

（1）新建一个 M 函数文件，实现将华氏温度转化为摄氏温度的功能。输入如下程序：

```
Function c = f2cfun (f)
```



```
c = 5 * (f - 32)/9;
c
```

（2）将此 M 文件命名为 f2cfun. m。

（3）调用 f2cfun. m 文件，计算 f = 100 时的结果。在命令窗口中输入：>> f2cfun（100）（回车），运行结果：________________。

4. 编写分段函数 $f(x)=\begin{cases} x+1, & -1\leqslant x<0 \\ 1, & 0\leqslant x<1 \\ x^2, & 1\leqslant x\leqslant 2 \end{cases}$ 的函数文件，存放于文件 ff. m 中，计算 f( -0.7)，f(0.5)，f(1.5)。

（1）ff. m 源程序：

（2）调用 ff. m，求 $f$（-0.7），$f$（0.5），$f$（1.5）。

$f$（-0.7）= ________________。

$f$（0.5）= ________________。

$f$（1.5）= ________________。

5. 给出一个百分制成绩，要求输出成绩等级'A'、'B'、'C'、'D'、'E'。90 分以上为'A'，81～89 分为'B'，70～79 分为'C'，60～69 分为'D'，60 分以下为'E'。要求：用户从键盘分别输入 5 个等级的百分制成绩，然后根据程序运行的结果验证所设计程序的正确性；当用户输入小于 0 的数时，结束程序。

（1）源程序：

（2）运行程序，分别用 94，86，78，60，45 作为测试数据，写出运行结果。

输入：94

输出：________________。

输入：86

输出：________________。

输入：78

输出：________________。

输入：60

输出：________________。

输入：45

输出：________________。

6. 用循环语句编写函数 jc. m 求 n!

（1）源程序：

（2）运行程序，分别用 1，10，22 作为测试数据，写出运行结果。

输入：1

输出：________________。

输入 ：10

输出：________________。

输入：22

输出：________________。

7. 设 $S=2+4+\cdots+2n$，用循环语句求 S 不超过 1 000 的 n 的最大值，并求出此时 S 的值。

（1）程序：

（2）运行结果：

S = ________________。

n = ________________。

8. 建立一个命令 M－文件：求所有的“水仙花数”，所谓“水仙花数”是指一个三位数，其各位数字的立方和等于该数本身。例如，153 是一个水仙花数，因为 $153=1^3+5^3+3^3$。

（1）程序：

（2）运行结果：

9. （选做）猜数游戏。

首先，由计算机随机产生一个［1，100］之间的一个整数，然后由用户猜测所产生的这个数。根据用户猜测的情况给出不同的提示，如果猜测的数大于产生的数，则显示“High”，小于则显示“Low”，等于则显示“You won!”，同时退出游戏。用户最多有 7 次机会，每次用户猜错时，提示还剩的次数，若 7 次后仍未猜对，显示“You lost!”。



程序：

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 命令文件与函数文件的区别有哪些？

2. 调试程序的过程中用了哪些方法？

3. 实验小结。

書

实验四

# MATLAB 的符号计算

## （验证性实验）

### 一、实验目的

1. 熟悉符号变量的定义与性质。
2. 掌握符号变量的使用与运算过程。
3. 掌握 MATLAB 中常见的符号运算命令。

### 二、实验原理

1. 符号对象的建立：sym 函数和 syms 语句。

（1）sym 函数：用来建立单个符号变量或表达式，一般调用格式为：

符号变量 = sym（A）。

（2）syms 语句：syms 语句一次可以定义多个符号变量。一般调用格式为：

syms　符号变量名 1　符号变量名 2 … 符号变量名 n

用这种格式定义符号变量时变量间用空格而不要用逗号分隔。

2. 符号变量确定原则。

（1）除了 i 和 j 之外，字母位置最接近 x 的字母；若距离相等，则取 ASCII 码大的作为默认的符号变量；

（2）若没有除了 i 与 j 以外的字母，则视 x 为默认的符号变量。

3. MATLAB 的常见符号运算命令。

factor（s）：对符号表达式 s 分解因式。

expand（s）：对符号表达式 s 进行展开。

collect（s）：对符号表达式 s 合并同类项。

collect（s，v）：对符号表达式 s 按变量 v 合并同类项。

simplify（s）：应用函数规则对 s 进行化简。

simple（s）：调用 MATLAB 的其他函数对表达式进行综合化简，并显示化简过程。

findsym（s，n）：查找一个符号表达式中的符号变量，函数返回符号表达式 s 中的 n 个符号变量，若没有指定 n，则返回 s 中的全部符号变量。

subs（f，x，a）：用给定的数据替换符号表达式中的指定的符号变量。

4. 符号微积分。

（1）符号极限。

limit 函数的调用格式为：

- limit（f，x，a）：求符号函数 f(x) 的极限值。即计算当变量 x 趋近于常数 a 时，f(x)函数的极限值。
- limit（f，a）：求符号函数 f(x) 的极限值。由于没有指定符号函数 f(x) 的自变量，则使用该格式时，符号函数 f(x) 的变量为函数 findsym(f) 确定的默认自变量，即变量 x 趋近于 a。
- limit（f）：求符号函数 f(x) 的极限值。符号函数 f(x) 的变量为函数 findsym（f）确定的默认变量；没有指定变量的目标值时，系统默认变量趋近于 0，即 a=0 的情况。
- limit（f，x，a，'right'）：求符号函数 f 的右极限值。'right'表示变量 x 从右边趋近于 a。
- limit（f，x，a，'left'）：求符号函数 f 的左极限值。'left'表示变量 x 从左边趋近于 a。

例：
```
syms a m x;
f=(sin (a+x)-sin (a-x))/x;
limit (f)
limit (f, inf)        %求f函数在x→∞（包括+∞和-∞）处的极限
limit (f, x, inf, 'left')        %求f函数在x→∞的极限
```

（2）符号导数。

diff 函数用于对符号表达式求导数。该函数的一般调用格式为：

diff（s）：没有指定变量和导数阶数，则系统按 findsym 函数指示的默认变量对符号表达式 s 求一阶导数。

diff（s，v）：以 v 为自变量，对符号表达式 s 求一阶导数。

diff（s，n）：按 findsym 函数指示的默认变量对符号表达式 s 求 n 阶导数，n 为正整数。

diff（s，v，n）：以 v 为自变量，对符号表达式 s 求 n 阶导数。

例：
```
syms x;
f=sqrt (1+exp (x));
diff (f)        %未指定求导变量和阶数，按缺省规则处理，默认对x求一阶导数
diff (f, x, 2)        %对x求二阶导数
```

（3）符号积分。

符号积分由函数 int 来实现。该函数的一般调用格式为：

int（s）：没有指定积分变量和积分阶数时，系统按 findsym 函数指示的默认变量对被积函数或符号表达式 s 求不定积分。

int（s，v）：以 v 为自变量，对被积函数或符号表达式 s 求不定积分。

int（s，v，a，b）：求定积分运算。a，b 分别表示定积分的下限和上限。该函数求被积



函数在区间［a，b］上的定积分。a 和 b 可以是两个具体的数，也可以是一个符号表达式，还可以是无穷（inf）。当函数 f 关于变量 x 在闭区间［a，b］上可积时，函数返回一个定积分结果。当 a，b 中有一个是 inf 时，函数返回一个广义积分。当 a，b 中有一个符号表达式时，函数返回一个符号函数。

例：x = sym（'x'）；

f = (3 - x^2)^3；

f1 = int（f）

f2 = int（f，2，3）

（4）级数符号求和。

求无穷级数的和需要符号表达式求和函数 symsum，其调用格式为：

symsum（s，v，n，m）

其中 s 表示一个级数的通项，是一个符号表达式。v 是求和变量，v 省略时使用系统的默认变量。n 和 m 是求和的开始项和末项。

例：n = sym（'n'）；

s1 = symsum（1/n^2，n，1，inf）

（5）函数的泰勒级数。

MATLAB 提供了 taylor 函数将函数展开为幂级数，其调用格式为：

taylor（f，v，n，a）

该函数将函数 f 按变量 v 展开为泰勒级数，展开到第 n 项（即变量 v 的 n - 1 次幂）为止，n 的缺省值为 6。v 的缺省值与 diff 函数相同。参数 a 指定将函数 f 在自变量 v = a 处展开，a 的缺省值是 0。

例：x = sym（'x'）；

f1 = (1 + x + x^2)/(1 - x + x^2)；

taylor（f1，x，5）　　%展开到 x 的 4 次幂时应选择 n = 5

taylor（f1，x，5，1）　　%在 x = 1 处泰勒展开式的前 5 项

## 三、实验内容

1. 符号对象的建立和使用。
2. 符号计算中的算符和基本函数。
3. 符号表达式和符号函数的操作。
4. MATLAB 求极限的命令与计算方法。
5. MATLAB 求微分的命令与计算方法。
6. MATLAB 求积分的命令与计算方法。
7. MATLAB 求级数的和的命令与计算方法。
8. MATLAB 符号代数方程求解的命令与计算方法。
9. MATLAB 符号常微分方程求解的命令与计算方法。





# 四、实验报告

# MATLAB 符号运算

实验名称：________________ 实验日期：______年____月____日

姓名：________________ 班级学号：________________

成绩：________________

## （一）实验内容及步骤

1. 输入以下程序，观察 a1，a2，a3，a4 取值的差异。

```
a1 = [1/3, pi/7, sqrt(5), pi + sqrt(5)]
a2 = sym([1/3, pi/7, sqrt(5), pi + sqrt(5)])
a3 = sym([1/3, pi/7, sqrt(5), pi + sqrt(5)], 'r')
a4 = sym('[1/3, pi/7, sqrt(5), pi + sqrt(5)]')
```

运行：>> a24 = a2 - a4

运行结果：________________________________。

a2 与 a4 是否相等？为什么会出现这种结果？

2. 对独立自由符号变量的自动辨认。

（1）输入如下程序：

```
syms a b x X Y;
k = sym('3');
z = sym('c * sqrt(delta) + y * sin(theta)');
EXPR = a * z * X + (b * x^2 + k) * Y;
```

（2）输入 >> findsym（EXPR）

运行结果：________________________________。

（3）输入 >> findsym（EXPR，1）

运行结果：________________________________。

（4）输入 >> findsym（EXPR，2），findsym（EXPR，3）



运行结果：________________________________________。

3. 输入以下程序，体会按不同的方式合并同幂项的方法。

```
EXPR = sym ('(x^2 + x * exp (-t) + 1) * (x + exp (-t))');
expr1 = collect (EXPR)
expr2 = collect (EXPR, 'exp (-t)')
```

运行结果：________________________________________。

4. factor 指令的使用

(1)
```
syms a x;
f1 = x^4 - 5 * x^3 + 5 * x^2 + 5 * x - 6;
factor (f1)
```

运行结果：________________________________________。

(2)
```
f2 = x^2 - a^2;
factor (f2)
```

运行结果：________________________________________。

(3)
```
factor (546);
```

运行结果：________________________________________。

5. 简化 $f = \sqrt[3]{\frac{1}{x^3} + \frac{6}{x^2} + \frac{12}{x} + 8}$

(1) 程序：

(2) 运行结果：________________________________________。

6. 求下列函数的极限：

(1) $\lim\limits_{x \to 0} \frac{\cos x - e^{-\frac{x^2}{2}}}{x^4}$

程序：

运行结果：________________________________________。

(2) $\lim\limits_{x \to +\infty} \left(1 + \frac{2t}{x}\right)^{3x}$

程序：



运行结果：________________________________。

7. 按要求实现下面的符号求导运算：

（1）已知 $y=e^{2x}\ln(x^2+1)\tan(-x)$，求 y'，y"；

程序：

运行结果：________________________________。

（2）已知 $z=(x^2+y^2)e^{\frac{x^2+y^2}{xy}}$，求 $\frac{\partial z}{\partial x}$，$\frac{\partial^2 z}{\partial x^2}$，$\frac{\partial^2 z}{\partial x\partial y}$。

程序：

运行结果：________________________________。

8. 求下列符号积分：

（1）求不定积分 $\int x^3e^{-x^2}dx$；

程序：

运行结果：________________________________。

（2）求定积分：$\int_{\frac{\pi}{4}}^{\frac{\pi}{3}}\frac{x}{\sin^2x}dx$；

程序：

运行结果：________________________________。

（3）求二重积分：$\int_0^1\int_y^{\sqrt{y}}x\sin x\,dxdy$；

程序：



运行结果：________________________________________。

9. 试求解无穷级数的和 $S=\frac{1}{1\times4}+\frac{1}{4\times7}+\frac{1}{7\times10}+\cdots+\frac{1}{(3n-2)(3n+1)}+\cdots$

程序：

运行结果：________________________________________。

10. 试求出函数 $f(x)=\frac{\sin x}{(x^2+4x+3)}$ 的麦克劳林幂级数展开式的前 5 项，并求出关于 $x=2$ 的 Taylor 幂级数展开式的前 7 项。

程序：

运行结果：________________________________________。

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 可以用符号运算求出 $\int_0^1 \frac{\sin x}{x}dx$ 的解吗？为什么？

2. 实验小结。

书

实验五

# MATLAB的数值计算
（验证性实验）

## 一、实验目的

1. 掌握 MATLAB 矩阵分析的命令和方法。
2. 掌握 MATLAB 多项式运算的命令和方法。
3. 掌握 MATLAB 数值微积分的运算方法。

## 二、实验原理

1. 矩阵分析。

(1) 矩阵的转置。

转置运算符是单撇号（'）。

(2) 矩阵的旋转。

矩阵的旋转利用函数 rot90（A，k），功能是将矩阵 A 旋转 90°的 k 倍，当 k 为 1 时可省略。

(3) 矩阵的左右翻转。

对矩阵 A 实施左右翻转的函数是 fliplr（A）。

(4) 矩阵的上下翻转。

对矩阵 A 实施上下翻转的函数是 flipud（A）。

(5) 矩阵的逆。

求矩阵 A 的逆矩阵可调用函数 inv（A）。

(6) 矩阵的行列式。

求矩阵 A 所对应的行列式的值的函数是 det（A）。

(7) 矩阵的秩。

MATLAB 中，求矩阵秩的函数是 rank（A）。

(8) 矩阵的迹。

MATLAB 中，求矩阵的迹的函数是 trace (A)。

(9) 将矩阵转化成行最简形的命令是 rref (A)。

(10) 矩阵的特征值与特征向量。

MATLAB 中，计算矩阵 A 的特征值和特征向量的函数是 eig (A)，常用的调用格式有：

- E = eig (A)：求矩阵 A 的全部特征值，构成向量 E。
- [V, D] = eig (A)：求矩阵 A 的全部特征值，构成对角阵 D，并求 A 的特征向量构成 V 的列向量。

2. 多项式。

(1) 多项式的建立。

已知一个多项式的全部根 X，求多项式系数的函数是 poly(X)，该函数返回以 X 为全部根的一个多项式 P，当 X 是一个长度为 m 的向量时，P 是一个长度为 m + 1 的向量。

(2) 多项式求根。

求多项式 p(x) 的根的函数是 roots(P)，这里，P 是 p(x) 的系数向量，该函数返回方程 p(x) = 0 的全部根（含重根，复根）。

(3) 多项式求值。

求多项式 p(x) 在某点或某些点的函数值的函数是 polyval(P, x)。若 x 为一数值，则求多项式在该点的值；若 x 为向量或矩阵，则对向量或矩阵中的每个元素求其多项式的值。

例：
```
P =[1, -1, -6];
X = roots (P)              %求方程 p(x) =0 的根。
G = poly (X)               %求多项式 g (x)。
X0 =[2, 3];
f = polyval (P, X0)        %求多项式 p(x)在给定点的值。
```

(4) 多项式的四则运算。

- 多项式的加减法，采用运算符 + 和 - 。
- 多项式的乘法：函数 conv (P1, P2) 用于求多项式 P1 和 P2 的乘积。
- 多项式的除法：函数 [Q, r] = deconv (P1, P2) 用于对多项式 P1 和 P2 做除法运算。其中 Q 返回多项式 P1 除以 P2 的商式，r 返回 P1 除以 P2 的余式。这里，Q 和 r 仍是多项式系数向量。

deconv 是 conv 的逆函数，即有 P1 = conv (P2, Q) + r。

例：在 MATLAB 命令窗口，输入命令：
```
f =[4,6, -9,6];g =[2,1];g1 =[0,0,g];
f + g1                     %求 f(x)+ g(x)。
f - g1                     %求 f(x)- g(x)。
conv(f,g)                  %求 f(x) * g(x)。
[Q,r] = deconv(f,g)        %求 f(x)/g(x),商式存入 Q,余式存入 r。
```

3. 数值微积分。

(1) 数值微分。

MATLAB 中，没有直接提供求数值导数的函数，只有计算向前差分的函数。

- DX = diff (X)：计算向量 X 的向前差分，DX (i) = X (i+1) - X (i)，0 < i < n。
- DX = diff (X, n)：计算 X 的 n 阶向前差分，diff (X, 2) = diff (diff (X))。
- DX = diff (A, n, dim)：计算矩阵 A 的 n 阶差分，dim = 1 时（缺省状态），按列计算差分，dim = 2 时，按行计算差分。

(2) MATLAB 数值积分函数。

- 梯形法：trapz (x, y)，x 为分割点（节点）组成的向量，y 为被积函数在节点上的函数值组成的向量。

例：X = 0: 0.01: 1; Y = exp (-X.^2);
    trapz (X, Y)

- 抛物线法：quad (f, a, b, tol)，f = f (x) 为被积函数，[a, b] 为积分区间，tol 为计算精度，quad 适用于 f 的低阶数值积分。

另一函数：quad8 (f, a, b, tol)

用法与 quad 完全相同，适用于 f 的高阶数值积分。

例：用两种不同的方法求积分。

先建立一个函数文件 ex.m：

```
function ex = ex (x)
ex = exp (-x.^2);            % 注意应用点运算。
return
```

然后，在 MATLAB 命令窗口，输入命令：

```
quad ('ex', 0, 1, 1e-6)      % 注意函数名应加字符引号。
quad8 ('ex', 0, 1, 1e-6)     % 用另一函数求积分。
```

- 抛物线法计算二重积分：dblquad (f, a, b, c, d, tol)，用法与 quad 类似。

例：

```
g = inline ('exp (-x.^2 - y.^2) ');
dblquad (g, 0, 1, 0, 1)      % 直接调用二重积分函数求解。
```

## 三、实验内容

1. 特殊矩阵的建立和使用。
2. 求矩阵的转置、旋转、翻转、逆、方阵的行列式、秩、迹、特征值与特征向量。
3. 多项式的建立、求根、求值以及多项式的四则运算。
4. 数值微积分的计算。





# 四、实验报告

## MATLAB 数值运算

实验名称：______________________　　实验日期：________年____月____日
姓名：__________________________　　班级学号：______________________
成绩：__________________________

### （一）实验内容及步骤

1. 先建立 4 阶魔方矩阵 A，然后将 A 的第 1 行元素乘以 1，第 2 行乘以 2，……，第 4 行乘以 4。（提示：相当于矩阵 A 与一个对角阵相乘。）

程序：

运行结果：

2. 对以上魔方矩阵 A 实行结构变换。

（1）求 A 的转置。

运行结果：

（2）将 A 旋转 90°。

运行结果：



(3) 对 A 实行左右翻转。

运行结果：

(4) 对 A 实行上下翻转。

运行结果：

3. 求 $A=\begin{pmatrix}1 & 2 & 3\\ 2 & 2 & 1\\ 3 & 4 & 3\end{pmatrix}$的逆矩阵。

程序：

运行结果：

4. 计算方阵 $A=\begin{pmatrix}1 & 2 & 3\\ 2 & 2 & 1\\ 3 & 4 & 3\end{pmatrix}$的行列式的值。

程序：

运行结果：



5. 求向量组$\alpha_1=(1\ -2\quad 2\quad 3)$, $\alpha_2=(-2\quad 4\ -1\quad 3)$, $\alpha_3=(-1\quad 2\quad 0\quad 3)$, $\alpha_4=(0\quad 6\quad 2\quad 3)$, $\alpha_5=(2\quad -6\quad 3\quad 4)$的秩，并判断其线性相关性。

程序：

运行结果：

6. 求矩阵$A=\begin{pmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{pmatrix}$的迹。

程序：

运行结果：

7. 求矩阵$A=\begin{pmatrix}-2 & 1 & 1\\ 0 & 2 & 0\\ -4 & 1 & 3\end{pmatrix}$的特征值和特征向量。

程序：

运行结果：



8. 已知一个多项式 $P(x)=3x^5+4x^3-5x^2-7.3x+5$，计算：

（1）计算 $P(x)=0$ 的全部根。

程序：

运行结果：

（2）由方程 $P(x)=0$ 的根构造一个多项式 $g(x)$，并与 $P(x)$ 进行对比。

程序：

运行结果：

（3）计算 $P(-1)$、$P(2.4)$、$P(5)$、$P(6.8)$ 的值。

程序：

运行结果：

9. 设有两个多项式 $f(x)=3x^5-5x^4+2x^3-7x^2+5x+6$，$g(x)=3x^2+5x-3$ 计算：

（1）求 $f(x)+g(x)$、$f(x)-g(x)$。

（2）求 $f(x)\cdot g(x)$、$f(x)/g(x)$。

（1）程序：

运行结果：

（2）程序：



运行结果：

10. 求向量 sin(X) 的 1 ~3 阶差分。设 X 由 [0，2π] 间均匀分布的 10 个点组成。

程序：

```
X = linspace (0, 2 * pi, 10);
Y = sin (X);
DY = diff (Y);              %计算 Y 的一阶差分。
D2Y = diff (Y, 2);          %计算 Y 的二阶差分，也可用命令 diff (DY) 计算。
D3Y = diff (Y, 3);          %计算 Y 的三阶差分，也可用 diff (D2Y) 或 diff (DY, 2)。
```

运行结果：

DY = ______________________

D2Y = ______________________

D3Y = ______________________

11. 用梯形法和抛物线法近似计算 $\int_2^3 \frac{1}{\ln x} dx$。

(1) 梯形法。

程序：

运行结果：

(2) 抛物线法。

程序：

运行结果：



12. 计算$\int_{-1}^{1}\int_{-2}^{2}e^{-x^2/2}\sin(x^2+y)dxdy$的数值解。

程序：

运行结果：

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 求矩阵的逆是否只有 inv（）一种方法？还有其他的方法吗？你可以想出几种求矩阵逆的方法？

2. 通过定积分数值解的实验，是否掌握了求定积分数值解的原理和方法？你可以自己编程实现吗？

3. 实验小结。



# 第三部分

# 数学实验

## 实验六

# 代数方程的求解

（验证性实验）

## 一、实验目的

1. 掌握求解线性方程组的命令和方法。
2. 掌握求解代数方程的命令和方法。
3. 通过实例学习用线性代数方程组解决简化的实际问题。

## 二、实验原理

我们将线性方程的求解分为两类：一类是方程组求唯一解或求特解，另一类是方程组求

无穷解即通解。可以通过系数矩阵的秩来判断：

- 若系数矩阵的秩 r = n（n 为方程组中未知变量的个数），则有唯一解。
- 若系数矩阵的秩 r < n，则可能有无穷解。

线性方程组的无穷解 = 对应齐次方程组的通解 + 非齐次方程组的一个特解；其特解的求法属于解的第一类问题，通解部分属第二类问题。

1. 求线性方程组的唯一解或特解（第一类问题）

这类问题的求法分为两类：一类主要用于解低阶稠密矩阵 —— 直接法；另一类是解大型稀疏矩阵 —— 迭代法。

（1）利用矩阵除法求线性方程组的特解（或一个解）。

方程：AX = b

解法：X = A\b

（2）利用矩阵的 LU、QR 和 cholesky 分解求方程组的解。

- LU 分解：LU 分解又称 Gauss 消去分解，可把任意方阵分解为下三角矩阵的基本变换形式（行交换）和上三角矩阵的乘积。即 A = LU，L 为下三角阵，U 为上三角阵。

则：A * X = b　变成 L * U * X = b

所以 X = U\(L\b)，这样可以提高运算速度。

命令：[L，U] = lu（A）

- Cholesky 分解：若 A 为对称正定矩阵，则 Cholesky 分解可将矩阵 A 分解成上三角矩阵和其转置的乘积，即：A = R' * R，其中 R 为上三角阵。

方程 A * X = b 变成　R' * R * X = b

所以 X = R \ （R' \ b）

命令：R = chol（A）

- QR 分解：对于任何长方矩阵 A，都可以进行 QR 分解，其中 Q 为正交矩阵，R 为上三角矩阵的初等变换形式，即：A = QR

方程 A * X = b 变形成 QRX = b

所以 X = R\(Q\b)

命令：[Q，R] = qr（A）

说明：这三种分解，在求解大型方程组时很有用。其优点是运算速度快、可以节省磁盘空间、节省内存。

2. 求线性齐次方程组的通解

在 MATLAB 中，函数 null 用来求解零空间，即满足 A · X = 0 的解空间，实际上是求出解空间的一组基（基础解系）。

格式：z = null　　　　% z 的列向量为方程组的正交规范基，满足 Z' × Z = I。

z = null（A，'r'）　　　% z 的列向量为方程 AX = 0 的有理基。

3. 求非齐次线性方程组的通解

非齐次线性方程组需要先判断方程组是否有解，若有解，再去求通解。因此，步骤为：

第一步：判断 AX = b 是否有解，若有解则进行第二步。

第二步：求 AX = b 的一个特解。

第三步：求 AX = 0 的通解。



第四步：AX = b 的通解 = AX = 0 的通解 + AX = b 的一个特解。

4. 符号代数方程求解

在 MATLAB 中，求解用符号表达式表示的代数方程可由函数 solve 实现，其调用格式为：

solve (s)：求解符号表达式 s 的代数方程，求解变量为默认变量。

solve (s，v)：求解符号表达式 s 的代数方程，求解变量为 v。

solve (s1，s2，…，sn，v1，v2，…，vn)：求解符号表达式 s1，s2，…，sn 组成的代数方程组，求解变量分别 v1，v2，…，vn。

## 三、实验内容

1. 求线性方程组的唯一解或特解。
2. 分别利用 LU 分解、Cholesky 分解、QR 分解求方程组的解。
3. 求线性齐次方程组的通解。
4. 求非齐次线性方程组的通解。
5. 符号代数方程的求解。





# 四、实验报告

## 代数方程（组）的求解

实验名称：______________ 实验日期：______年____月____日

姓名：______________ 班级学号：______________

成绩：______________

### （一）实验内容及步骤

1. 求方程组 $\begin{cases} x_1-2x_2+3x_3-4x_4=4 \\ x_2-x_3+x_4=-3 \\ x_1+3x_2+x_4=1 \\ -7x_2+x_3+x_4=-3 \end{cases}$ 的解。

（1）建立系数矩阵 A 和常数项矩阵 B。

A = ______________________________________________。

B = ______________________________________________。

（2）求增广矩阵的秩。

R_A = ______________________________________________。

（3）对方程组解的情况进行判定。

方程组解的判定为：______________（无解、有唯一解或有无穷多解）

（4）求解方程组。

程序：

运行结果：



2. 求解方程组的通解：

$$\begin{cases} x_1 + 2x_2 + x_3 - x_4 = 0 \\ 3x_1 + 6x_2 - x_3 - 3x_4 = 0 \\ 5x_1 + 10x_2 + x_3 - 5x_4 = 0 \end{cases}$$

（1）建立系数矩阵 A，并判断解的情况。

程序：

结论：

（2）求齐次线性方程组的基础解系。

```
format rat
B = null（A，'r'）%求解空间的有理基。
```

运行后显示结果如下：

（3）写出通解：

```
syms  k1  k2
X = k1 * B（:，1）+ k2 * B（:，2）  %写出方程组的通解。
```

运行结果：

3. 求解方程组：

$$\begin{cases} x_1 - 2x_2 + 3x_3 - x_4 + 2x_5 = 2 \\ 3x_1 - x_2 + 5x_3 - 3x_4 + x_5 = 6 \\ 2x_1 + x_2 + 2x_3 - 2x_4 - x_5 = 8 \end{cases}$$

（1）建立系数矩阵 A 和增广矩阵 B，并求它们的秩。（分别用变量 R－A 和 R－B 表示）

程序：

（2）对解进行判定并求解。

```
format rat
if R_A == R_B&R_A == n  %判断有唯一解
```



```
        X = A\b
    elseif R_A == R_B&R_A < n    %判断有无穷解
        X = A\b    %求特解
        C = null (A, 'r') %求 AX = 0 的基础解系
        else
        X = 'The equition has no solution!'    %判断无解
    end
```

运行结果：

4. 请自己编程完成求解方程组的通解。

$$\begin{cases} x_1 + x_2 - 3x_3 - x_4 = 1 \\ 3x_1 - x_2 - 3x_3 + 4x_4 = 4 \\ x_1 + 5x_2 - 9x_3 - 8x_4 = 0 \end{cases}$$

程序：

运行结果：

写出通解：

5. 求代数方程：$ax^2 - bx - 6 = 0$ 的解。（提示：用 solve 命令求解）

程序：



运行结果：

6. 求代数方程组$\begin{cases}2x+3y=0\\4x^2+3y=1\end{cases}$的解。

程序：

运行结果：

7. （选做）用 LU 分解、QR 分解、Cholesky 分解求解第 1 题中的方程组。

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 第 4 题的求解还有其他方法吗？你能想出几种方法？

2. 实验小结。

# 实验七

# 微分方程的求解

## （验证性实验）

## 一、实验目的

1. 学会用 MATLAB 软件用数值和解析两种方法求解微分方程。
2. 通过实例学习用微分方程解决实际问题。

## 二、实验原理

1. 符号常微分方程求解。

常微分方程有时很难求解，MATLAB 提供了功能强大的工具，可以帮助求解微分方程。函数 dsolve 计算常微分方程的符号解。dsolve 语句中用字母 D 来表示求微分，D2，D3 等表示重复求微分，并以此来设定方程。任何 D 后所跟的字母为因变量。独立变量可以指定或由 symvar 规则选定为缺省。dsolve 函数也可用来求解微分方程组，其调用格式为：

```
y = dsolve('eq1','eq2', ... ,'cond1','cond2', ... ,'v')
```

其中 y 为输出，eq1、eq2、... 为微分方程，cond1、cond2、... 为初值条件，v 为自变量。

例如，一阶方程 dy/dx = 1 + y2 的通解为：

```
>> dsolve ('Dy = 1 + y^2')
   ans =
         - tan ( - x + C1)
```

其中，C1 是常数。若上述方程中加入初值 y（0）= 1，求其特解，则可改为：

```
>> dsolve ('Dy = 1 + y^2 ', 'y (0) = 1')
   ans =
        tan (x + 1/4 * pi)
```

独立变量可用如下形式指定：



```
>> dsolve ('Dy=1+y^2', 'y(0)=1', 'v')          % 指定微分变量为 v
       ans =
              tan(v+1/4*pi)
```

接下来举一个二阶微分方程的例子，该方程有两个初始条件：

d2y = cos (2x) − y (0) = 0 y (0) = 1，y' (0) = 0

```
>> y=dsolve('D2y=cos(2*x)-y', 'Dy(0)=0', 'y(0)=1')
   y =
      -2/3*cos(x)^2+1/3+4/3*cos(x)
>> y=simple(y) % 化简
   y =
      -1/3*cos(2*x)+4/3*cos(x)
```

2. 常微分方程的数值解

在微分方程难以获得解析解的情况下，可以求其数值解。其调用格式为：

```
[T, Y] =solver(odefun, tspan, y0)
```

说明：

(1) 其中的 solver 为命令 ode45、ode23、ode113、ode15s、ode23s、ode23t、ode23tb 之一。

(2) odefun 是显式常微分方程：$\begin{cases} \dfrac{dy}{dt} = f(t, y) \\ y(t_0) = y_0 \end{cases}$。

(3) 在积分区间 tspan =[$t_0$，$t_f$] 上，从 $t_0$ 到 $t_f$，用初始条件 $y_0$ 求解。

(4) 要获得问题在其他指定时间点 $t_0$，$t_1$，$t_2$，…上的解，则令 tspan =[$t_0$，$t_1$，$t_2$，…，$t_f$]（要求是单调的）。

(5) 因为没有一种算法可以有效地解决所有的 ODE 问题，为此，MATLAB 提供了多种求解器 solver，对于不同的 ODE 问题，采用不同的 solver（如表 3-7-1 所示）。

**表 3-7-1**

| 求解器 Solver | ODE 类型 | 特　点 | 说　明 |
|---|---|---|---|
| ode45 | 非刚性 | 单步算法；4、5 阶 Runge-Kutta 方程；累计截断误差达 $(\Delta x)^3$ | 大部分场合的首选算法 |
| ode23 | 非刚性 | 单步算法；2、3 阶 Runge-Kutta 方程；累计截断误差达 $(\Delta x)^3$ | 使用于精度较低的情形 |
| ode113 | 非刚性 | 多步法；Adams 算法；高低精度均可到 $10^{-3} \sim 10^{-6}$ | 计算时间比 ode45 短 |
| ode23t | 适度刚性 | 采用梯形算法 | 适度刚性情形 |
| ode15s | 刚性 | 多步法；Gear's 反向数值微分；精度中等 | 若 ode45 失效时，可尝试使用 |
| ode23s | 刚性 | 单步法；2 阶 Rosebrock 算法；低精度 | 当精度较低时，计算时间比 ode15s 短 |
| ode23tb | 刚性 | 梯形算法；低精度 | 当精度较低时，计算时间比 ode15s 短 |

（6）要特别提到的是：ode23、ode45 是极其常用的用来求解非刚性的标准形式的一阶常微分方程（组）的初值问题的解的 MATLAB 的常用程序，其中：

- ode23：采用龙格 - 库塔 2 阶算法，用 3 阶公式作误差估计来调节步长，具有低等的精度。
- ode45：则采用龙格 - 库塔 4 阶算法，用 5 阶公式作误差估计来调节步长，具有中等的精度。

3. inline（）：建立一个内联函数。格式：inline（'expr', 'var1', 'var2', …），注意括号里的表达式要加引号。

例 1：求解微分方程初值问题$\begin{cases}\dfrac{dy}{dx}=-2y+2x^2+2x\\ y(0)=1\end{cases}$的数值解，求解范围为区间[0, 0.5]。

```
fun = inline ('-2 * y + 2 * x^2 + 2 * x', 'x', 'y');
[x, y] = ode23 (fun, [0, 0.5], 1);
x';
y';
plot (x, y, 'o -')
>> x'
ans =
      0.0000   0.0400   0.0900   0.1400   0.1900   0.2400
      0.2900   0.3400   0.3900   0.4400   0.4900   0.5000
>> y'
ans =
      1.0000   0.9247   0.8434   0.7754   0.7199   0.6764
      0.6440   0.6222   0.6105   0.6084   0.6154   0.6179
```

图形结果如图 3 -7 -1。

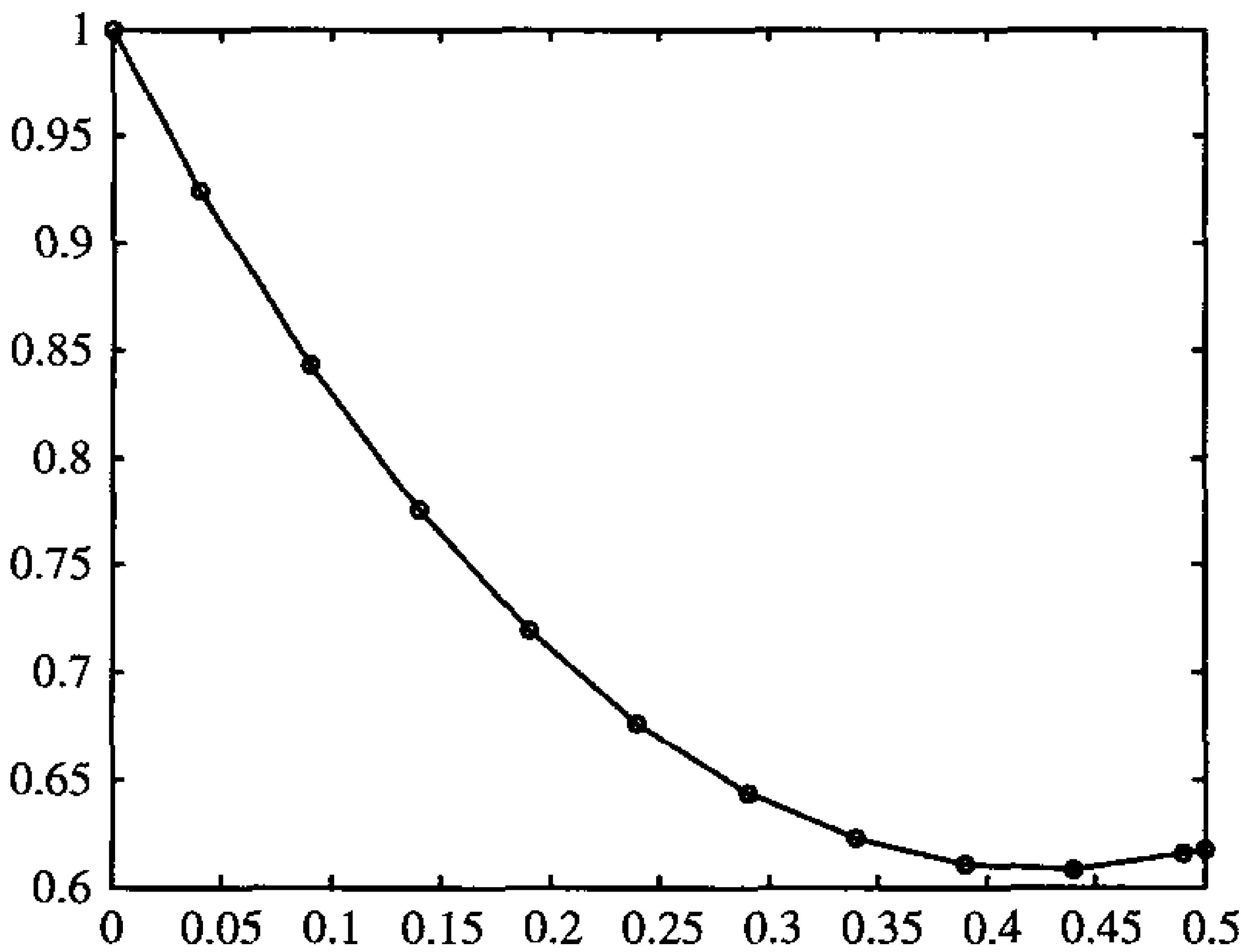



**图 3 -7 -1**

例 2：求解描述振荡器的经典的 Ver der Pol 微分方程

$\frac{d^2y}{dt^2}-\mu(1-y^2)\frac{dy}{dt}+y=0$，$y(0)=1$，$y'(0)=0$，$\mu=7$。

分析：令 $x_1=y$，$x_2=\frac{dx_1}{dt}$，则 $\frac{dx_1}{dt}=x_2$，$\frac{dx_2}{dt}=\mu(1-x_1^2)x_2-x_1$。

先编写函数文件 verderpol. m：

```
function xprime = verderpol (t, x)
global mu;
xprime =[x (2); mu * (1 - x (1) ^2) * x (2) - x (1)];
```

再编写命令文件 vdp1. m：

```
global mu;
mu = 7;
y0 =[1; 0]
[t, x] = ode45 ('verderpol', [0, 40], y0);
x1 = x (:, 1); x2 = x (:, 2);
plot (t, x1)
```

图形结果如图 3 -7 -2。

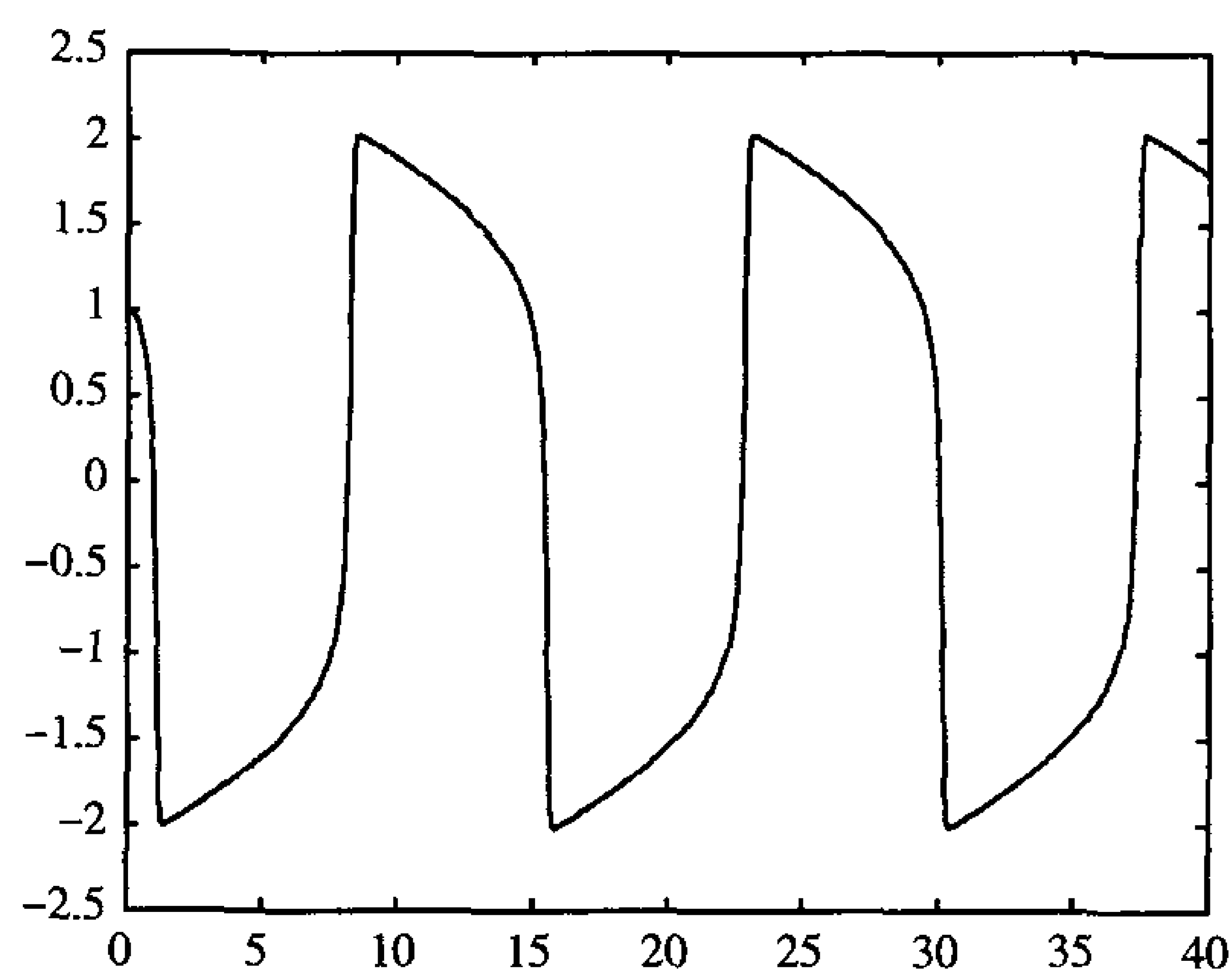



**图 3 -7 -2**

## 三、实验内容

1. 利用 dsolve 命令求微分方程的解析解。
2. 求微分方程的数值解。



# 四、实验报告

## 微分方程的求解

实验名称：____________　　实验日期：______年____月____日
姓名：____________　　班级学号：____________
成绩：____________

### （一）实验内容及步骤

1. 求微分方程$\frac{dy}{dx}+y\tan x=\cos^2 x$的解析解，并加以验证。

（1）方程的求解。

程序：

运行结果：

（2）验证。

程序：



结论：

2. 求微分方程 $y''=\sin 2x-y$ 在初始条件 $y(0)=0$，$y'(0)=1$ 下的特解，并画出解函数的图形。

程序：

运行结果：

3. 求微分方程组 $\begin{cases}\dfrac{dx}{dt}+x+y=0\\ \dfrac{dy}{dt}+x-y=0\end{cases}$ 在初始条件 $x|_{t=0}=1$，$y|_{t=0}=0$ 下的特解，并画出解函数的图形。

程序：

运行结果：

4. 求解微分方程初值问题 $\begin{cases}\dfrac{dy}{dx}=2y+x+2\\ y(0)=1\end{cases}$ 的数值解，求解范围为区间 [0, 1]。

程序：

运行结果：



5. 求解微分方程初值问题$\begin{cases}\dfrac{dy}{dx}=\dfrac{3}{x}y+x^3\ (e^x+\cos x)\ -2x\\ y\mid_{x=\pi}=(e^{\pi}+\dfrac{2}{\pi})\ \pi^3\end{cases}$的数值解，求解范围为区间 $[\pi,\ 2\pi]$，并画出其数值解和解析解曲线。

(1) 求解析解。

程序：

运行结果：

(2) 求数值解。

程序：

运行结果：

(3) 画出解析解和数值解的图形。

程序：

运行结果：



6. 用 ode23（或 ode45）求解初值问题$\frac{d^2y}{dx^2}+2\frac{dy}{dx}+y=\cos x$，y（0）=0，y'（0）=$\frac{3}{2}$的数值解。

分析：令 $y_1=y$，$y_2=\frac{dy_1}{dx}$，则$\frac{dy_1}{dx}=y_2$，$\frac{dy_2}{dx}=-2y_2-y_1+\cos(x)$。

（1）编写函数文件 fun. m，作为 ode45 输入函数。

程序：

（2）求微分方程的数值解，并画出图形。

程序：

运行结果：

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 比较第 5 题中方程的解析解和数值解。结果相差大吗？实验结果是否正确？

2. 实验小结。

实验八

# 非线性方程求解
## （设计性实验）

## 一、实验目的

1. 掌握用 MATLAB 软件求解非线性方程和方程组的基本用法，并对结果作出初步的分析。

2. 练习用非线性方程组建立实际问题的模型并进行求解。

## 二、实验原理

1. 单变量非线性方程求解。

在 MATLAB 中提供了一个 fzero 函数，可以用来求单变量非线性方程的根。对于求方程 f(x)=0 的近似解，该函数的调用格式为：

- z = fzero（f，x0）　　　　%求函数 f（x）在点 x0 附近的零点 z
- z = fzero（f，[a，b]）　　%求函数 f（x）在区间（a，b）内的零点 z

例：求 f（x）= x − 1/x + 5 在 x0 = −5 附近的零点。

先编制一个函数文件 fz. m：

```
function f = f（x）
f = x - 1/x + 5;
```

然后，在 MATLAB 命令窗口，输入命令：

```
fzero（'fz'，-5）　　　　%以 -5 作为迭代初值。
```

Zero found in the interval：[ −4. 8，−5. 2]。

2. 非线性方程组的求解。

对于非线性方程组 F（X）=0，用 fsolve 函数求其数值解。fsolve 函数的调用格式为：

```
[x，fval] = fsolve（'fun'，x0，option）
```

其中 x 为返回的解，fval 是解向量处对应的函数值向量。fun 是用于定义需求解的非线



性方程组的函数文件名，x0 是求根过程的初值，fsolve 所求得的解是最接近初始值的解。option 为最优化工具箱的选项设定。最优化工具箱提供了 20 多个选项，用户可以使用 optimset 命令将它们显示出来。如果想改变其中某个选项，则可以调用 optimset（）函数来完成。例如，Display 选项决定函数调用时中间结果的显示方式，其中 ‘off’ 为不显示，‘iter’表示每步都显示，‘final’ 只显示最终结果。optimset（‘Display’，‘off’）将设定 Display 选项为 ‘off’。

例：求下列非线性方程组 $\begin{cases} x-0.6\sin x-0.3\cos y=0 \\ y-0.6\cos x+0.3\sin y=0 \end{cases}$ 在（0.5，0.5）附近的数值解。

（1）建立函数文件 myfun. m。

```
function q = myfun (p)
x = P(1);
y = P(2);
q (1) = x - 0.6 * sin (x) - 0.3 * cos (y);
q (2) = y - 0.6 * cos (x) + 0.3 * sin (y);
```

（2）在给定的初值 x0 = 0.5，y0 = 0.5 下，调用 fsolve 函数求方程的根。

```
x = fsolve ('myfun', [0.5, 0.5] ', optimset ('Display', 'off'))
x =
    0.6354
    0.3734
```

将求得的解代回原方程，可以检验结果是否正确，命令如下：

```
q = myfun (x)
q =
   1.0e - 009 *
   0.2375    0.2957
```

可见得到了较高精度的结果。

## 三、实验内容

1. 利用 fzero 命令求单变量非线性方程的解。
2. 利用 fsolve 命令求非线性方程组的解。



# 四、实验报告

## 非线性方程的求解

实验名称：______________________　　实验日期：________年____月____日
姓名：__________________________　　班级学号：______________________
成绩：__________________________

## （一）实验内容及步骤

1. 求方程 $10x + e^x - 2 = 0$ 的根。
（1）画出曲线图形，确定根所在区间。
程序：

运行结果：

（2）利用 fzero（）函数求解方程的根。
程序：

运行结果：



2. 求方程组$\begin{cases}\sin x+y+z^2e^x-4=0\\x+yz=0\\xyz=0\end{cases}$在（1，1，1）附近的解并对结果进行验证。

（1）编写函数文件 fun1. m，建立方程左边的向量值函数。

程序：

（2）调用 fsolve（）函数求解方程组。

程序：

运行结果：

3. 求平面 $3x+5y+6z=0$ 和平面 $x-3y-6z-1=0$ 的交线与球面 $x^2+y^2+z^2=9$ 的交点。

分析：要求两平面的交线与球面的交点，将它们联立成方程组求解即可。

（1）编写函数文件 fun2. m，建立方程左边的向量值函数。

程序：

（2）求解方程组，得到交点。

程序：

运行结果：

4. 小张夫妇以按揭方式贷款买了一套价值20万元的房子，首付5万元，每月还款1 000元，15年还清。问贷款利率是多少?

分析：假设初始需付的款为 $a_0$，$a_1$，…，$a_n$ 分别表示第1年到第n年需付款的余额。每年付款b元，经n年后还清，付款利率为r，于是对于按揭付款的方式可列出下列方程：

$$a_1 = a_0 * (1 + r) - b$$
$$a_2 = a_1 * (1 + r) - b$$
$$\vdots$$
$$a_n = a_{n-1} * (1 + r) - b$$

于是有

$$\begin{aligned} a_n &= a_{n-2}(1 + r)^2 - b - b(1 + r) \\ &= a_{n-3}(1 + r)^3 - b - b(1 + r) - b(1 + r)^2 \\ &\vdots \\ &= a_0 * (1 + r)^n - b(1 + (1 + r) + (1 + r)^2 + \cdots + (1 + r)^{n-1}) \\ &= a_0 * (1 + r)^n - b(1 + (1 + r) + (1 + r)^2 + \cdots + (1 + r)^{n-1}) \\ &= a_0 * (1 + r)^n - b((1 + r)^n - 1)/r \end{aligned}$$

若按月还款，则将上式的n换为12n，年利率换为月利率，年付款额换为月付款额即可。按照上述的关系建立函数文件 rate1. m：

```
function y = rate1(x)
        y = 150 * (1 + x)^(15 * 12) - 1 * ((1 + x)^(15 * 12) - 1)/x;   %计算月利率
```

调用 fzero ( ) 函数求解方程的根：

```
[x1,fv1]= fzero(@rate1,1)
```

运行结果：

结论：

5. 某人贷款50万元购房，他咨询了两家银行，第一家银行开出的条件是每月还4 500元，15年还清；第二家银行开出的条件是每年还45 000元，20年还清。从利率方面看，哪家银行较优惠（简单地假设年利率=月利率×12）?

由上题的分析易得：

$$a_n = a_0 * (1 + r)^n - b\ ((1 + r)^n - 1)\ /r$$

(1) 编写函数文件 rate2. m. 及 rate3. m，分别用来保存计算两家银行利率的方程。



程序：

（2）调用 fzero（）函数分别求两个方程的根。
程序：

运行结果：

结论：应选择哪家银行贷款？

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. fsolve 求解非线性方程组时，得到的解是最接近初始值的解，所以一次只能得到一组解。若方程有多组解，应该如何处理？你能想到几种方法？

2. 实验小结。

书

# 实验九

# 线性规划问题的求解
## （设计性实验）

## 一、实验目的

1. 掌握用 MATLAB 优化工具箱解线性规划问题的方法。
2. 练习建立实际问题的线性规划模型并求解。

## 二、实验原理

线性规划（Linear Programming）是运筹学的一个重要的分支，它的应用十分广泛，不仅许多实际问题属于线性规划问题，而且运筹学的其他分支的一些问题也可以转化为线性规划问题，因此，线性规划问题求解在最优化中占据重要地位。

MATLAB 中求解标准形式的线性规划问题的命令是 linprog，常用的有以下几种形式：

1. 模型：$\min z = cX$

s. t. $AX \leqslant b$

命令：x = linprog（c，A，b）。

2. 模型：$\min z = cX$

s. t. $AX \leqslant b$

$Aeq \cdot X = beq$

命令：x = linprog（c，A，b，Aeq，beq）。

注意：若没有不等式 $AX \leqslant b$ 存在，则令 A =[ ]，b =[ ]。

3. 模型：$\min z = cX$

s. t. $AX \leqslant b$

$Aeq \cdot X = beq$



$$VLB \leqslant X \leqslant VUB$$

命令：x = linprog（c，A，b，Aeq，beq，VLB，VUB）。

注意：若没有不等式 $AX \leqslant b$ 存在，则令 A =[ ]，b =[ ]，若没有等式约束 $Aeq \cdot X = beq$，则令 Aeq =[ ]，beq =[ ]。

4. 命令：[x，fval] = linprog（c，A，b，Aeq，beq，VLB，VUB）。

返回最优解 x 及 x 处的目标函数值 fval。

例：求解线性规划问题

$$\max z = 0.4x_1 + 0.28x_2 + 0.32x_3 + 0.72x_4 + 0.64x_5 + 0.6x_6$$

$$\begin{aligned} \text{s.t.}\quad & 0.01x_1 + 0.01x_2 + 0.01x_3 + 0.03x_4 + 0.03x_5 + 0.03x_6 \leqslant 850 \\ & 0.02x_1 + 0.05x_4 \leqslant 700 \\ & 0.02x_2 + 0.05x_5 \leqslant 100 \\ & 0.03x_3 + 0.08x_6 \leqslant 900 \\ & x_j \geqslant 0 \qquad j = 1, 2, \cdots, 6 \end{aligned}$$

由于题中的目标函数为求最大值，非线性规划的标准形，因此首先将其化为标准形：

$$\min z = -0.4x_1 - 0.28x_2 - 0.32x_3 - 0.72x_4 - 0.64x_5 - 0.6x_6$$

求解程序如下：

```
c =[ -0.4  -0.28  -0.32  -0.72  -0.64  -0.6];
A =[0.01 0.01 0.01 0.03 0.03 0.03;0.02 0 0 0.05 0 0;0 0.02 0 0 0.05 0;0 0 0.03 0 0 0.08];
b =[850;700;100;900];
Aeq =[ ]; beq =[ ];
vlb =[0;0;0;0;0;0]; vub =[ ];
[x,fval]= linprog(c,A,b,Aeq,beq,vlb,vub)
```

## 三、实验内容

1. 通过对实际问题的分析建立线性规划模型。
2. 利用 MATLAB 的 linprog（）函数求解线性规划问题。



# 四、实验报告

## 线性规划问题的求解

实验名称：________________ 实验日期：________年____月____日

姓名：________________ 班级学号：________________

成绩：________________

### （一）实验内容及步骤

1. 求下列线性规划模型的解。

$$\max\ z=\frac{3}{4}x_1-150x_2+\frac{1}{50}x_3-6x_4$$

$$\text{s.t.}\quad \frac{1}{4}x_1-60x_2-\frac{1}{50}x_3+9x_4\leqslant 0$$

$$-\frac{1}{2}x_1+90x_2+\frac{1}{50}x_3-3x_4\geqslant 0$$

$$x_1\geqslant -5,\ x_2\geqslant -5,\ -5\leqslant x_3\leqslant 1,\ x_4\geqslant -5$$

（1）该模型是否为标准形？若不是，先将其化为标准形。

（2）依模型，生成 linprog（）函数的各项参数。

程序：

（3）调用 linprog（）函数求解。

程序：



运行结果：

该模型的解为：

目标函数的值为：

2. 某工厂生产每件产品需经 A，B，C 三个车间，每个车间所需的工时数如表 3－9－1 所示。

**表 3－9－1**

| 车　　间 | A | B | C |
|---|---|---|---|
| 生产单位甲产品需工时数 | 2 | 1 | 0 |
| 生产单位乙产品需工时数 | 1 | 1 | 1 |
| 一周可用工时数 | 10 | 8 | 7 |

已知生产单位甲产品工厂每周可获利 4 万元，生产单位乙产品工厂每周可获利 3 万元，问该厂如何安排生产才能使每周获得的利润最大？

（1） 建立线性规划模型。

（2） 求解建立的模型。

程序：

运行结果：

结论：

3. 炼油厂将 A、B、C 三种原油加工成甲、乙、丙三种汽油。一桶原油加工成汽油的费用为 4 元，每天至多能加工汽油 14 000 桶。原油的买入价、买入量、辛烷值、硫含量，及汽油的卖出价、需求量、辛烷值、硫含量由表 3－9－2 和表 3－9－3 给出。

（1）问如何安排生产计划，在满足需求的条件下使利润最大？

（2）一般来说，做广告可以增加销售，估计一天向一种汽油投入一元广告费，可以使这种汽油日销量增加 10 桶。问如何安排生产计划和广告计划使利润最大？

**表 3－9－2**

| 原油类别 | 买入价（元/桶） | 买入量（桶/天） | 辛烷值（%） | 硫含量（%） |
|---|---|---|---|---|
| A | 45 | ≤5 000 | 12 | 0.5 |
| B | 35 | ≤5 000 | 6 | 2.0 |
| C | 25 | ≤5 000 | 8 | 3.0 |

**表 3－9－3**

| 汽油类别 | 卖出价（元/桶） | 需求量（桶/天） | 辛烷值（%） | 硫含量（%） |
|---|---|---|---|---|
| 甲 | 70 | 3 000 | ≥10 | ≤1.0 |
| 乙 | 60 | 2 000 | ≥8 | ≤2.0 |
| 丙 | 50 | 1 000 | ≥6 | ≤1.0 |

（1）建立模型。

A 种原油加工的汽油桶数分别为 $x_{11}$，$x_{12}$，$x_{13}$。

B 种原油加工的汽油桶数分别为 $x_{21}$，$x_{22}$，$x_{23}$。

C 种原油加工的汽油桶数分别为 $x_{31}$，$x_{32}$，$x_{33}$。

依题意知，目标函数为总利润，记为 f，约束条件为买入量、需求量的限制，加工能力的限制以及辛烷值、硫含量的要求。可得目标函数为：

约束条件：

每天至多加工汽油总量为 14 000 桶：



A、B、C 的买入量都小于 5 000 桶：

甲、乙、丙三种汽油日需求总量分别为 3 000 桶、2 000 桶、1 000 桶：

甲、乙、丙的辛烷值有要求：

甲、乙、丙的硫含量有要求：

加工汽油量为非负数：

模型的求解：
程序：

运行结果：



结论：应如何安排生产计划？每天的最大利润为多少？

（2）下面考虑投入广告费用的情况：
A 种原油加工的汽油桶数分别为 $x_{11}$，$x_{12}$，$x_{13}$。
B 种原油加工的汽油桶数分别为 $x_{21}$，$x_{22}$，$x_{23}$。
C 种原油加工的汽油桶数分别为 $x_{31}$，$x_{32}$，$x_{33}$。
对甲汽油的广告投资 $y_1$，对乙汽油的广告投资 $y_2$，对丙汽油的广告投资 $y_3$。
同理分析得，目标函数为：

约束条件：
每天至多加工汽油总量为 14 000 桶：

A、B、C 的买入量都小于 5 000 桶：

甲、乙、丙三种日需求量还要加上广告效应：

甲、乙、丙的辛烷值有要求：

甲、乙、丙的硫含量有要求：



加工汽油量为非负数，广告费也为非负数：

程序：

运行结果：

结论：如何安排生产？是否应该投人广告费用？每天的最大利润为多少？

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 求解线性规划问题时，如遇到变量的取值必须取整数怎么办？

2. 实验小结。

实验十

# 数据的统计描述和分析
## （验证性实验）

## 一、实验目的

1. 掌握数据的统计描述和参数估计、假设检验的基本概念与原理，及用 MATLAB 实现的方法。

2. 练习用这些方法解决实际问题。

## 二、实验原理

1. 基本统计处理。

最大值和最小值。MATLAB 提供的求数据序列的最大值和最小值的函数分别为 max 和 min，两个函数的调用格式和操作过程类似。

① 求向量的最大值和最小值。

求一个向量 X 的最大值的函数有两种调用格式，分别是：

- y = max（X）：返回向量 X 的最大值存入 y，如果 X 中包含复数元素，则按模取最大值。
- [y，I] = max（X）：返回向量 X 的最大值存入 y，最大值的序号存入 I，如果 X 中包含复数元素，则按模取最大值。

求向量 X 的最小值的函数是 min（X），用法和 max（X）完全相同。

例：求向量 x 的最大值。

命令如下：

```
x =[ -43, 72, 9, 16, 23, 47];
y = max (x)             %求向量 x 中的最大值。
[y, l] = max (x)        %求向量 x 中的最大值及其位置。
```

② 求矩阵的最大值和最小值。

求矩阵 A 的最大值的函数有 3 种调用格式，分别是：

- max（A）：返回一个行向量，向量的第 i 个元素是矩阵 A 的第 i 列上的最大值。
- [Y，U] = max（A）：返回行向量 Y 和 U，Y 向量记录 A 的每列的最大值，U 向量记录每列最大值的行号。
- max（A，[ ]，dim）：dim 取 1 或 2。dim 取 1 时，该函数和 max（A）完全相同；dim 取 2 时，该函数返回一个列向量，其第 i 个元素是 A 矩阵的第 i 行上的最大值。

求最小值的函数是 min，其用法和 max 完全相同。

③ 两个向量或矩阵对应元素的比较。

函数 max 和 min 还能对两个同型的向量或矩阵进行比较，调用格式为：

- U = max（A，B）：A，B 是两个同型的向量或矩阵，结果 U 是与 A，B 同型的向量或矩阵，U 的每个元素等于 A，B 对应元素的较大者。
- U = max（A，n）：n 是一个标量，结果 U 是与 A 同型的向量或矩阵，U 的每个元素等于 A 对应元素和 n 中的较大者。

*min* 函数的用法和 *max* 完全相同。

2. 求和与求积。

数据序列求和与求积的函数是 sum 和 prod，其使用方法类似。设 X 是一个向量，A 是一个矩阵，函数的调用格式为：

sum（X）：返回向量 X 各元素的和。

prod（X）：返回向量 X 各元素的乘积。

sum（A）：返回一个行向量，其第 i 个元素是 A 的第 i 列的元素和。

prod（A）：返回一个行向量，其第 i 个元素是 A 的第 i 列的元素乘积。

sum（A，*dim*）：当 *dim* 为 1 时，该函数等同于 sum（A）；当 dim 为 2 时，返回一个列向量，其第 i 个元素是 A 的第 i 行的各元素之和。

prod（A，dim）：当 dim 为 1 时，该函数等同于 prod（A）；当 dim 为 2 时，返回一个列向量，其第 i 个元素是 A 的第 i 行的各元素乘积。

3. 平均值和中值。

求数据序列平均值的函数是 mean，求数据序列中值的函数是 median。两个函数的调用格式为：

mean（X）：返回向量 X 的算术平均值。

median（X）：返回向量 X 的中值。

*mean*（A）：返回一个行向量，其第 i 个元素是 A 的第 i 列的算术平均值。

median（A）：返回一个行向量，其第 i 个元素是 A 的第 i 列的中值。

mean（A，dim）：当 dim 为 1 时，该函数等同于 mean（A）；当 dim 为 2 时，返回一个列向量，其第 i 个元素是 A 的第 i 行的算术平均值。

median（A，dim）：当 dim 为 1 时，该函数等同于 median（A）；当 dim 为 2 时，返回一个列向量，其第 i 个元素是 A 的第 i 行的中值。

4. 累加和与累乘积。

在 MATLAB 中，使用 cumsum 和 cumprod 函数能方便地求得向量和矩阵元素的累加和与累乘积向量，函数的调用格式为：

cumsum（X）：返回向量 X 累加和向量。

cumprod（X）：返回向量 X 累乘积向量。

cumsum（A）：返回一个矩阵，其第 i 列是 A 的第 i 列的累加和向量。

cumprod（A）：返回一个矩阵，其第 i 列是 A 的第 i 列的累乘积向量。

cumsum（A，dim）：当 dim 为 1 时，该函数等同于 cumsum（A）；当 dim 为 2 时，返回一个矩阵，其第 i 行是 A 的第 i 行的累加和向量。

cumprod（A，dim）：当 dim 为 1 时，该函数等同于 cumprod（A）；当 dim 为 2 时，返回一个向量，其第 i 行是 A 的第 i 行的累乘积向量。

5. 标准方差与相关系数。

（1）求标准方差。在 MATLAB 中，提供了计算数据序列的标准方差的函数 std。对于向量 X，std（X）返回一个标准方差。对于矩阵 A，std（A）返回一个行向量，它的各个元素便是矩阵 A 各列或各行的标准方差。std 函数的一般调用格式为：

Y = std(A,flag,dim)

其中 dim 取 1 或 2。当 dim = 1 时，求各列元素的标准方差；当 dim = 2 时，则求各行元素的标准方差。flag 取 0 或 1，当 flag = 0 时，按 σ1 所列公式计算标准方差，当 flag = 1 时，按 σ2 所列公式计算标准方差。缺省 flag = 0，dim = 1。

（2）相关系数。MATLAB 提供了 corrcoef 函数，可以求出数据的相关系数矩阵。corrcoef 函数的调用格式为：

- corrcoef（X）：返回从矩阵 X 形成的一个相关系数矩阵。此相关系数矩阵的大小与矩阵 X 一样。它把矩阵 X 的每列作为一个变量，然后求它们的相关系数。
- corrcoef（X，Y）：在这里，X，Y 是向量，它们与 corrcoef（[X，Y]）的作用一样。

例：生成满足正态分布的 10 000 ×5 随机矩阵，然后求各列元素的均值和标准方差，再求这 5 列随机数据的相关系数矩阵。

命令如下：

```
X = randn（10 000，5）；
M = mean（X）
D = std（X）
R = corrcoef（X）
```

6. 排序。

MATLAB 中对向量 X 是排序函数是 sort（X），函数返回一个对 X 中的元素按升序排列的新向量。sort 函数也可以对矩阵 A 的各列或各行重新排序，其调用格式为：

[Y，I] = sort（A，dim）

其中 dim 指明对 A 的列还是行进行排序。若 dim = 1，则按列排；若 dim = 2，则按行排。Y 是排序后的矩阵，而 I 记录 Y 中的元素在 A 中位置。

7. 概率密度函数 pdf 系列。以 normpdf（ ）为例，调用格式：

y = normpdf（x，mu，sigma），

计算参数为 mu 和 sigma 的样本数据 x 的正态概率密度函数。参数 sigma 必须为正。其中：mu 为均值，sigma 为标准差。

8. 参数估计 fit 系列。以 normfit（）为例，调用格式：

[muhat, sigmahat, muci, sigmaci] = normfit (x, alpha),

对样本数据 x 进行参数估计，并计算置信度为 100 (1 - alpha)% 的置信区间，如 alpha = 0.01 时，则给出置信度为 99% 的置信区间。不写明 alpha，即表示 alpha 取 0.05。

9. hist (x, m) 函数：画样本数据 x 的直方图，m 为直方图的条数，缺省值为 10。

10. ttest (x, m, alpha) 函数：假设检验函数。此函数对样本数据 x 进行显著性水平为 alpha 的 t 假设检验，以检验正态分布样本 x（标准差未知）的均值是否为 m。h = 1 表示拒绝零假设，h = 0 表示不能拒绝零假设。

11. normplot (x) 或 weibplot (x) 函数：统计绘图函数，进行正态分布检验，如果数据是来自一个正态分布，则该线为一直线形态；如果它是来自其他分布，则为曲线形态。

## 三、实验内容

1. 求数据的最大最小值、中值、均值、标准差和相关系数。
2. 对数据进行参数估计。
3. 进行正态假设检验。



# 四、实验报告

## 数据的统计描述与分析

实验名称：____________________　　实验日期：_______年___月___日

姓名：____________________　　班级学号：____________________

成绩：____________________

## （一）实验内容及步骤

1. 假定一个月 31 天的三城市每日最高气温被记录，并赋给脚本 M 文件中的变量 temps，在 MATLAB 工具箱里取名为 temp. m。

temps = [12,8,18;15,9,22;12,5,19;14,8,23;12,6,22;11,9,19;15,9,5;8,10,20;19,7,18;12,7,18;14,10,19;11,8,17;9,7,23;8,8,19;15,8,18;8,9,20;10,7,17;12,7,22;9,8,19;12,8,21;12,8,20;10,9,17;13,12,18;9,10,20;10,6,22;14,7,21;12,5,22;13,7,18;15,10,23;13,11,24;12,12,22]

每一行包含了给定一天的高温；每一列包含不同城市的高温。

（1）运行 temp. m 文件，把变量 temps 放在 MATLAB 工作空间里。

（2）为了使数据可视，把它绘图：

程序：

```
plot（temps）
xlabel（'日期'），ylabel（'温度（℃）'）
title（'三个城市的日最高温度'）
```

（3）分别求出三个城市的最高温及最高温出现的日期。

程序：

运行结果：



（4）分别求出三个城市的最低温及最低温出现的日期。

程序：

运行结果：

（5）分别计算三个城市这个月温度的中值和平均值。

程序：

运行结果：

（6）计算三个城市的总平均温度。

程序：

运行结果：

（7）分别计算三个城市这个月温度的标准差。

程序：

运行结果：

（8）求三个城市温度之间的相关系数。

程序：

运行结果：

2. 某厂从一台机床生产的滚珠中随机抽取9个，测得直径（mm）如下：14.6，14.7，15.1，14.9，14.8，15.0，15.1，15.2，14.8。设滚珠直径服从正态分布，试自行给出不同的显著性水平，对直径的均值和标准差做区间估计。

（1）取显著性水平 $\alpha=0.05$。

程序：

运行结果见表3-10-1。

**表3-10-1**

| 点估计 | | 区间估计 | |
|---|---|---|---|
| 均值 | 标准差 | 均值 | 标准差 |
| | | | |

（2）取不同的显著性水平进行计算比较，见表3-10-2。

**表3-10-2**

| 显著性水平 α | 0.005 | 0.01 | 0.05 | 0.1 | 0.5 |
|---|---|---|---|---|---|
| 均值点估计 | | | | | |
| 均值区间估计 | | | | | |
| 标准差点估计 | | | | | |
| 标准差区间估计 | | | | | |

3. 检验上题中滚珠直径数据的正态性。

（1）利用统计绘图函数 normplot（x）进行分布的正态性检验，由于：

程序：



运行结果：

结论：是否服从正态分布？

（2）利用函数 ttest（x，m，alpha）进行显著性水平 $\alpha = 0.05$ 的 t 假设检验，并说明方程未知的情况下，均值 m = 15 是否合理？m = 16 呢？

程序：

运行结果：

结论：

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 根据第 2 题的实验结果，判断参数估计会随着显著性水平的变化发生怎样的变化？为什么会呈现这种变化？

2. 实验小结。

# 实验十一

# 曲线的插值与拟合
## （设计性实验）

## 一、实验目的

1. 掌握用 MATLAB 计算拉格朗日、分段线性、三次样条三种插值的方法，改变节点的数目，对三种插值结果进行初步分析。

2. 掌握 MATLAB 多项式拟合及最小二乘拟合命令的用法。

3. 通过实例学习用曲线插值和拟合解决实际问题。

## 二、实验原理

在大量应用中，人们常面临用一个解析函数描述数据间的关系问题。解决这个问题有两种方法。函数插值与曲线拟合都是要根据一组数据构造一个函数作为近似，由于近似的要求不同，二者的数学方法上是完全不同的。而面对一个实际问题，究竟用插值还是拟合，有时容易确定，有时则并不明显。在插值法里，数据假定是正确的，要求以某种方法描述数据点之间所发生的情况。在曲线拟合或回归方法里，是设法找出某条光滑曲线，使它最佳地拟合数据，而不必经过任何数据点。

1. 一维插值。

插值问题的提法是，已知 n+1 个节点 $(x_j, y_j)$，$j=0, 1, 2, \cdots, n$，其中 $x_j$ 互不相同，不妨设 $a=x_0<x_1<\cdots<x_n=b$，求任一插值点 $x^*$ $(\neq x_j)$ 处的插值 $y^*$。$(x_j, y_j)$ 可以看成是由某个函数 $y=g(x)$ 产生的，g 的解析表达式可能十分复杂，或不存在封闭形式，也可以未知。求解的基本思路是，构造一个相对简单的函数 $y=f(x)$，使 f 通过全部节点，即 $f(x_j)=y_j$ $(j=0, 1, 2, \cdots, n)$，再由 $f(x)$ 计算插值，即 $y^*=f(x^*)$。

MATLAB 中用 interp1（）函数来进行一维插值计算，其调用格式为：

y = interp1（x0，y0，x，'method'）

其中 x0，y0 是同维数据向量，分别表示插值节点的横、纵坐标，x 是待求函数值的插

值节点向量。'method'为可选项，说明插值使用的方法。对于一维插值 interp1 来说，MATLAB 提供的可选的方法有：nearest，linear，spline，cubic，它们分别表示最近插值，线性插值，三次样条插值和三次插值。命令返回值 y 是插值曲线在节点向量 x（横坐标）处的纵坐标向量。

三次样条插值命令：

```
y = interp1 (x0, y0, x, 'spline')
```

或

```
y = spline (x0, y0, x)
```

其中输入 x0，y0，x 和输出 y 的意义同上。

为说明一维插值，考虑下列问题，12 小时内每一小时测一次室外温度。数据存储在两个变量中。

```
hours = 1:12;
temps =[5,8,9,15,25,29, 31,30,22,25,7,24];      %记录的温度数值
plot(hours,temps,hours,temps,' + ')
title(' Temperature ')
xlabel(' Hour '),ylabel(' Degrees Celsius ')
```

为计算任给时间的温度，可用函数 interp1 作插值运算。

```
t = interp1(hours,temps,9.3)      %估计第 9.3 小时处的温度值
t =22.9000
t = interp1(hours, temps, 4.7)      %估计第 4.7 小时处的温度值
t =22
t = interp1(hours, temps,[3.2 6.5 7.1 11.7])
t =10.2000
30.0000
30.9000
24.9000
```

若不采用直线连接数据点，我们可采用某些更光滑的曲线来拟合数据点。最常用的方法是用三次样条插值。

```
t = interp1 (hours, temps, 9.3, ' spline ')
t =21.8577
t = interp1 (hours, temps, 4.7, ' spline ')
t =22.3143
t = interp1 (hours, temps, [3.2 6.5 7.1 11.7], ' spline ')
t =9.6734
30.0427
31.1755
25.3820
```

注意，样条插值得到的结果，与上面所示的线性插值的结果不同。因为插值是一个估计或猜测的过程，其意义在于应用不同的估计规则导致不同的结果。

2. 二维插值。

二维插值是基于与一维插值同样的基本思想，是对两个变量的函数 z = f（x，y）进行插值。

MATLAB 中用函数 interp2 来拟合二维网格（X，Y）上的数据 Z，语法是：

z1 = interp2（x，y，z，x1，y1，'method'）

其中（x，y，z）是已给的数据点的横、纵、竖坐标，（x1，y1）是插值点的横、纵坐标，z1 为插值点的竖坐标。'method' 为插值方法，主要有：

'linear'：线性插值，默认。

'pchip'：逐段三次 Hermite 插值。

'spline'：逐段三次样条函数插值。

其中最后一种插值的曲面比较平滑。

例：设有一平板，在均匀分布的格栅上采集温度值。数据如下：

```
width = 1:5;      % 平板宽度的分格,(x 方向)
depth = 1:3;      % 平板深度的分格,(y 方向)
temps =[82 81 80 82 84; 79 63 61 65 81; 84 84 82 85 86]      % 温度数据
temps =
82 81 80 82 84
79 63 61 65 81
84 84 82 85 86
```

如同在标引点上测量一样，矩阵 temps 表示整个平板的温度分布。temps 的列与下标 depth 或 y－维相联系，行与下标 width 或 x－维相联系。为了估计在中间点的温度，我们必须对它们进行辨识。

```
wi = 1:0.2:5;      % 中间点的宽度
d = 2;      % 中间点的深度
zlinear = interp2(width,depth,temps,wi,d);
zcubic = interp2(width, depth, temps, wi,d, ' cubic ');
plot(wi, zlinear, ' - ' , wi, zcubic)
xlabel(' Width of Plate '), ylabel(' Degrees Celsius ')
title( [' Temperature at Depth =' num2str(d) ] )
```

3. 曲线拟合。

设（$x_1$，$y_1$），（$x_2$，$y_2$），…，（$x_n$，$y_n$）是直角平面坐标系下给出的一组数据，设 $x_1 < x_2 < \cdots < x_n$，且已知它们满足某一函数 y = f（a，x），其中 a 是待定的参数，曲线拟合就是要确定这些参数，使得拟合值与真实值之间的误差达到最小。最常用的判断曲线拟合效果好坏的方法就是最小二乘法，即使得目标函数 sum（（yi－f（a，xi））^2）达到最小。当最佳拟合被解释为在数据点的最小误差平方和，且所用的曲线限定为多项式时，那么曲线拟合是相当简洁的。

（1）多项式拟合。

MATLAB 中进行多项式拟合的函数为 polyfit（）。调用格式：

```
p = polyfit(x,y,n)
```

```
[p,s]=polyfit(x,y,n)
```

说明：x，y 为数据点，n 为多项式阶数，返回 p 为幂次从高到低的多项式系数向量 p。矩阵 s 用于生成预测值的误差估计。

如

```
x=[0 .1 .2 .3 .4 .5 .6 .7 .8 .9 1];
y=[ -0.447 1.978 3.28 6.16 7.08 7.34 7.66 9.56 9.48 9.30 11.2];
p=polyfit(x, y, 2)
p= -9.8108 20.1293  -0.0317
```

可得其解为 $y=-9.8108x^2+20.1293x-0.0317$。

为了将曲线拟合解与数据点比较，可将二者都绘成图。

```
xi=linspace (0, 1, 100);
z=polyval (p, xi);%为计算在 xi 数据点的多项式值，调用 MATLAB 的函数 polyval
plot(x, y, 'o', x, y, xi, z, ':')
```

多项式阶次的选择是任意的。两点决定一直线或一阶多项式。三点决定一曲线或 2 阶多项式。按此进行，n+1 数据点唯一地确定 n 阶多项式。于是对于有 11 个数据点的上例情况，可选一个高达 10 阶的多项式。然而，高阶多项式给出很差的数值特性，故不应选择比所需阶次高的多项式。拟合曲线阶次“越多就越好”的观念是不适用的。

（2）最小二乘拟合。

在 MATLAB 的最优化工具箱中提供了 lsqcurvefit（）函数，可以解决最小二乘曲线拟合的问题。该函数的调用格式为：

```
[a,fval]=lsqcurvefit(fun,a0,x,y)
```

其中：fun 为原型函数的 MATLAB 表示，可以是 M 文件定义的函数或 inline（）函数，a0 为最优化的初值，x，y 为原始输入输出数据向量，a 为返回的待定系数向量，fval 为在此待定系数下的目标函数的值。

例：由函数 $y=0.5e^{-0.1x}+0.8e^{-0.3x}\sin(2.5x)$ 生成一组数据 x 和 y

```
x=0:.1:10;
y=0.5*exp( -0.1*x)+0.8*exp( -0.3*x).*sin (2.5*x);
```

假设以上数据满足原型为 $y=a_1e^{-a_2x}+a_3e^{-a_4x}\sin(a_5x)$，其中，$a_i$ 为待定系数。采用最小二乘曲线拟合的目的就是获得这些待定系数的值，使得目标函数的值最小。根据已知的函数原型，可以编写出如下函数：

```
f=inline('a(1)*exp( -a(2)*x)+a(3)*exp( -a(4)*x).*sin (a (5) *x)', 'a', 'x');
```

建立起函数原型后，就可以由下面的语句得出待定的系数向量了。

```
[xx,res]=lsqcurvefit(f,[1,1,1,1,1],x,y);
xx'
res
```

绘制出拟合曲线与样本点的图形：

```
x1=0:0.01:10; y1=f(xx,x1); plot(x1,y1,x,y,'o')
```



## 三、实验内容

1. 一维插值方法的实现。
2. 二维插值方法的实现。
3. 多项式拟合命令的使用方法。
4. 最小二乘拟合命令的使用方法。





# 四、实验报告

## 曲线的插值与拟合

实验名称：________________ 实验日期：______年____月____日
姓名：________________ 班级学号：________________
成绩：________________

### （一）实验内容及步骤

1. 已知数据，见表 3－11－1。

**表 3－11－1**

| x | 0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| y | 0.3 | 0.5 | 1 | 1.4 | 1.6 | 1.9 | 0.6 | 0.4 | 0.8 | 1.5 | 2 |

画出用线性、三次样条和三次多项式插值所得［0，1］区间内的曲线图，并求当 $x_i$ = 0.25、0.35、0.45 时的 $y_i$ 的值。

（1）画曲线图。

程序：

运行结果：

（2）求 $x_i$ = 0.25、0.35、0.45 时的 $y_i$ 的值。

程序：



运行结果:

2. 已知某处山区地形选点测量坐标数据为:

x=0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5

y=0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6

海拔高度数据为:

z=89 90 87 85 92 91 96 93 90 87 82
92 96 98 99 95 91 89 86 84 82 84
96 98 95 92 90 88 85 84 83 81 85
80 81 82 89 95 96 93 92 89 86 86
82 85 87 98 99 96 97 88 85 82 83
82 85 89 94 95 93 92 91 86 84 88
88 92 93 94 95 89 87 86 83 81 92
92 96 97 98 96 93 95 84 82 81 84
85 85 81 82 80 80 81 85 90 93 95
84 86 81 98 99 98 97 96 95 84 87
80 81 85 82 83 84 87 90 95 86 88
80 82 81 84 85 86 83 82 81 80 82
87 88 89 98 99 97 96 98 94 92 87

(1) 画出其地貌图。

程序:

运行结果:

(2) 对数据插值加密形成地貌图，原始数据用小圆圈标出。(将程序补充完整)

程序:

```
xi=linspace(0,5,50);     %加密横坐标数据到50个
yi=linspace(0,6,80);     %加密纵坐标数据到60个
```

运行结果：

3. 由离散数据，见表 3－11－2。

**表 3－11－2**

| x | 0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| y | 0.3 | 0.5 | 1 | 1.4 | 1.6 | 1.9 | 0.6 | 0.4 | 0.8 | 1.5 | 2 |

用 3 阶多项式拟合数据，并将原始曲线与拟合曲线进行比较。

（1）程序：

运行结果：

（2）从图像上观察拟合的效果。

4. 已知数据可能满足 $y = ax + bx^2e^{-cx} + d$，求满足数据的最小二乘解 a，b，c，d 的值，并将原始曲线与拟合曲线进行比较。见表 3－11－3 和表 3－11－4。

**表 3－11－3**

| $x_i$ | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 |
|---|---|---|---|---|---|
| $y_i$ | 2.320 1 | 2.647 0 | 2.970 7 | 3.288 5 | 3.600 8 |

**表 3－11－4**

| $x_i$ | 0.6 | 0.7 | 0.8 | 0.9 | 1.0 |
|---|---|---|---|---|---|
| $y_i$ | 3.909 0 | 4.214 7 | 4.519 1 | 4.823 2 | 5.127 5 |

（1）建立函数文件 e11f1.m，用来存储函数 $y = ax + bx^2e^{-cx} + d$。

程序：

（2）用最小二乘拟合函数拟合数据。

程序：

运行结果：

5. 在农业生产试验研究中，对某地区土豆的产量与化肥的关系做了一实验，得到了氮肥、磷肥的施肥量与土豆产量的对应关系，见表3－11－5。

**表3－11－5**

| 氮施肥量（千克/公顷） | 0 | 34 | 67 | 101 | 135 | 202 | 259 | 336 | 404 | 471 |
|---|---|---|---|---|---|---|---|---|---|---|
| 土豆产量（千克） | 15.18 | 21.36 | 25.72 | 32.29 | 34.03 | 39.45 | 43.15 | 43.46 | 40.83 | 30.75 |
| 磷施肥量（千克/公顷） | 0 | 24 | 49 | 73 | 98 | 147 | 196 | 245 | 294 | 342 |
| 土豆产量（千克） | 33.46 | 32.47 | 36.06 | 37.96 | 41.04 | 40.09 | 41.26 | 42.17 | 40.36 | 42.73 |

根据表3－11－5数据分别给出土豆产量与氮、磷肥的关系式。

（1）画出土豆产量与氮施肥量的散点图，观察它们的大致图形确定多项式的阶数。

程序：

运行结果：

结论：从散点图判断土豆产量与氮、磷肥的关系式应该采用的模型。

（2）利用 MATLAB 对数据进行拟合。

程序：



运行结果：

结论：

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 进行多项式拟合时，是否多项式阶数越高拟合效果越好？

2. 实验小结。



书

## 实验十二

# 神经网络及其在数据拟合中的应用
（设计性实验）

## 一、实验目的

1. 了解神经网络的基本知识。
2. 学会用 MATLAB 神经网络工具箱进行数据拟合。
3. 通过实例学习 MATLAB 神经网络工具箱的应用。

## 二、实验原理

人工神经网络是在对复杂的生物神经网络研究和理解的基础上发展起来的。我们知道，人脑是由大约 $10^{11}$ 个高度互连的单元构成，这些单元称为神经元，每个神经元约有 $10^4$ 个连接。仿照生物的神经元，可以用数学方式表示神经元，引入人工神经元的概念，并由神经元的互联可以定义出不同种类的神经网络。

1. 神经网络的概念及结构。

单个人工神经元的数学表示形式如图 3－12－1 所示。其中，$x_1$，$x_2$，…，$x_n$ 为一组输入信号，它们经过权值 $w_i$ 加权后求和，再加上阈值 b，则得出 $u_i$ 的值。可以认为该值为输入信号与阈值所构成的广义输入信号的线性组合。该信号经过传输函数 f(·) 可以得出神经元的输出信号 y。

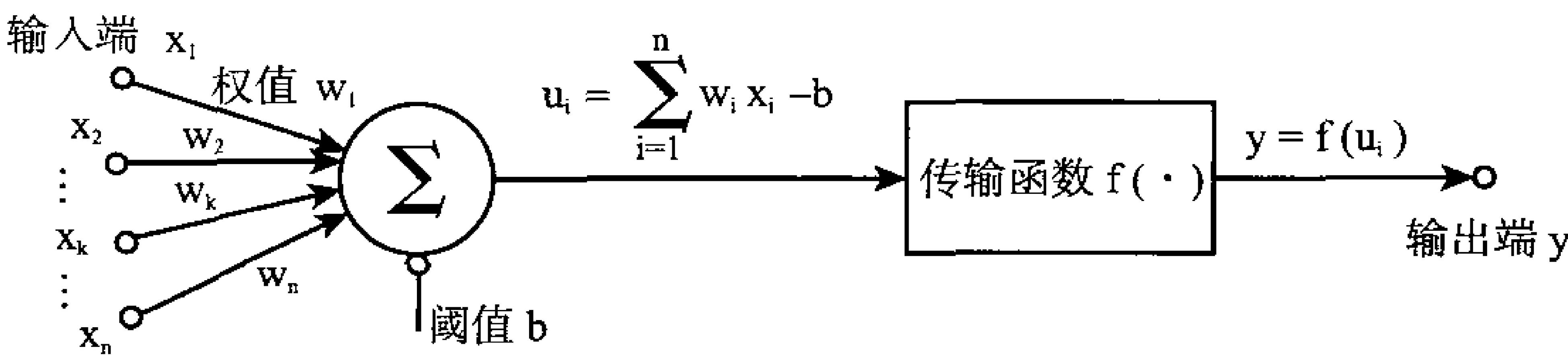



图 3－12－1

由若干个神经元相互连接，则可以构成一种网络，称为神经网络。由于连接方式的不同，神经网络的类型也不同。这里仅介绍前馈神经网络，因为其权值训练中采用误差逆向传播的方式，所以这类神经网络更多地称为反向传播（Back Propagation）神经网络，简称 BP 神经网络。BP 网的基本结构如图 3－12－2 所示。

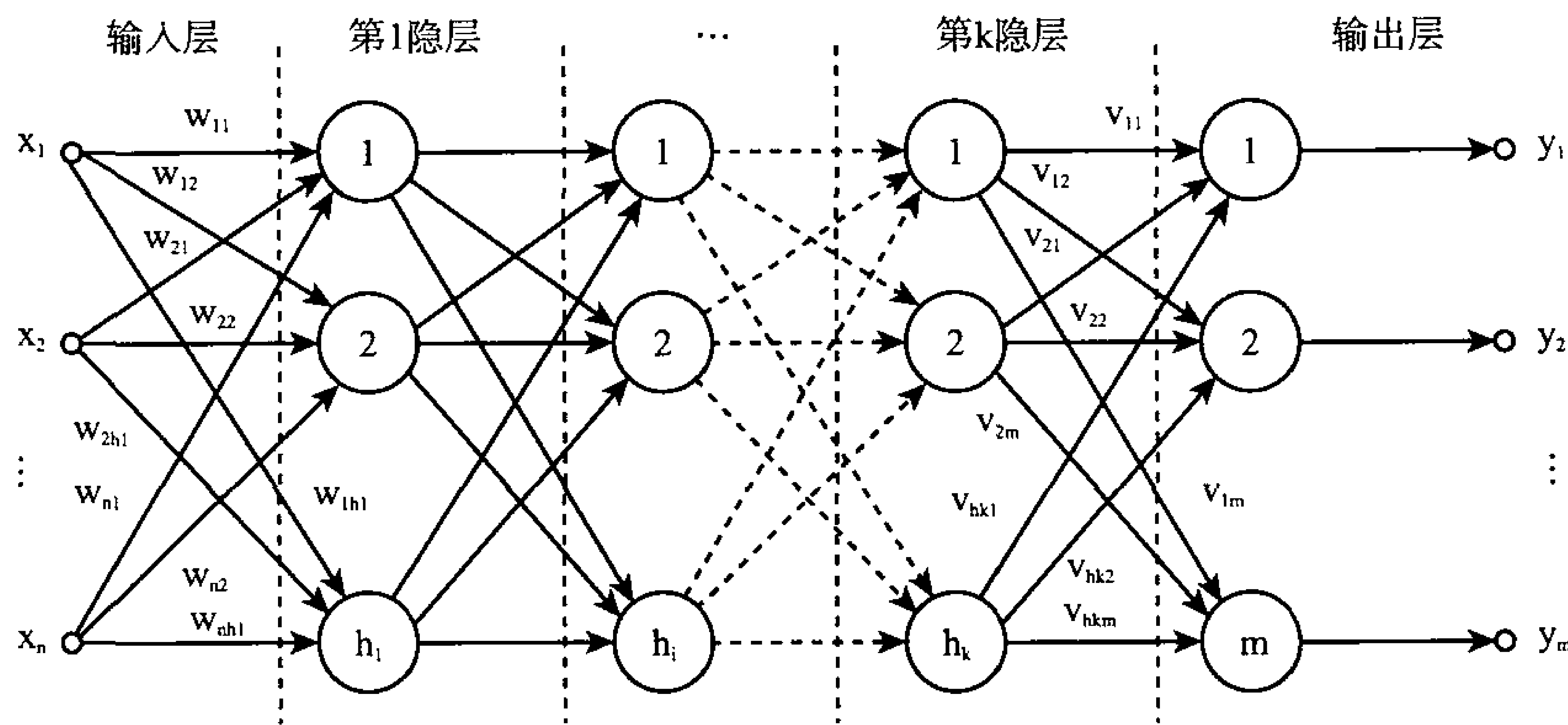



**图 3－12－2**

MATLAB 的神经网络工具箱提供了现成的函数和神经网络类，可以使用 newff（）函数来建立一个前馈的 BP 神经网络模型。newff（）的具体调用格式如下：

net = newff(x,y,[h1,h2,…,hk],{f1,f2,…,fk})

其中，x 为输入向量，y 为输出（目标）向量。[h1，h2，…，hk] 是一个行向量，用以存储神经网络各层的节点数，该向量的大小等于神经网络隐层的层数。{f1，f2，…，fk} 为一个元胞数组，由若干个字符串构成，每个字符串对应于该层的传输函数类型。当这些参数设定好后，就建立了一个神经网络数据对象 net，它的一些重要属性在表 3－12－1 给出。

**表 3－12－1**

| 属性名 | 属性说明 |
|---|---|
| net. IW | 输入层和隐层加权 |
| net. numInputs | 输入路数 |
| net. numLayers | 隐层数 |
| net. LW | 输入层和隐层加权 |
| net. trainParam. epochs | 最大训练步数 |
| net. trainParam. lr | 自学习的学习率 |
| net. trainParam. goal | 训练误差准则 |
| net. trainFcn | 训练算法 |

2. 神经网络的训练和泛化。若建立了神经网络模型 net，则可以调用 train（）函数对神经网络参数进行训练。该函数的调用格式为：

[net,tr,y1,e]= train(net,x,y)

其中，变量 x 为 $n \times M$ 的矩阵，n 为输入变量的维数，M 为样本的组数，y 为 $m \times M$ 的

矩阵，m 为输出变量的维数，x，y 分别存储样本点的输入和输出数据。由样本点数据进行训练，则可以得出训练后的神经网络对象 net，且可以返回其他相关的内容，tr 为结构体数据，返回训练的相关跟踪信息，tr. epochs 为训练步数，tr. perf 为各步目标函数的值。y1 和 e 矩阵分别返回由神经网络计算出的输出和误差矩阵。在训练过程中将每隔 25 步自动显示一次训练指标。训练结束后还可以用下面的语句绘制出目标值曲线：plotperf（tr）。

如果在给出的最大训练步数下无法得出满足要求的网络，则将给出错误的信息提示。用户可以再调用该函数一次，这时将以上次的训练结果加权矩阵为初值继续训练，用户可以循环调用该语句。如果误差在几次循环调用后仍无显著改善，则说明网络结构有问题，应该修改网络结构。

神经网络训练完成后，可以利用该网络对样本区域内的其他输入量求解其输出值，这种求值的方法称为神经网络的仿真或泛化（Generalization），可以理解为利用神经网络进行数据拟合，对新的输入点数据 x1 调用 sim（）函数进行泛化，得出这些输入点处的输出矩阵 y1，且 y1 = sim（net，x1）。

神经网络是否成功不在于对样本点本身拟合误差的大小，而关键在于其泛化效果。如果对样本点以外的其他输入点均有较好的拟合效果，则说明该神经网络结构合理。否则，训练出来的神经网络没有应用价值。

例：产生一组数据：

```
x = -1:0.2:1;y =1./(1 +25 * x.^2);
x0 = -1:0.1:1;y0 =1./(1 +25 * x0.^2);
```

其中，x，y 为训练数据，$x_0$，$y_0$ 为测试数据。由数据可知，输入变量 x 为一维数据，取值范围分别为［-1，1］。利用 newff（）函数建立 BP 神经网络。设定其有 2 个隐层，第 1 隐层有 5 个节点，该层神经元采用 tansig 传输函数，第 2 隐层含 1 个节点，传输函数为 tansig（）函数，建立神经网络模型：

```
net = newff(x,y,5,{'tansig'});
net = train(net,x,y);   %用 x，y 训练网络。
y1 = sim(net,x0);   %调用 sim（）函数进行泛化。
figure,plot(x,y,'o',x0,y0,x0,y1,':');   %从图形上看神经网络的泛化能力。
```

用神经网络对二元函数进行曲面拟合。

例：

```
[x,y]= meshgrid( -3:.6:3, -2:.4:2); x =x(:)'; y =y(:)';   %生成训练样本数据。
z =(x.^2 -2 * x). * exp( -x.^2 -y.^2 -x. * y);   % 注意 x，y，z 均应为行向量。
net = newff([x;y],z);   %二维输入，3 个隐层。
net.trainParam.epochs = 1000; net.trainFcn = 'trainlm';   %设定最大训练步数和训练算法。
[net,b]= train(net,[x; y],z);   % 训练神经网络。
[x2,y2]= meshgrid( -3:.1:3, -2:.1:2); x1 =x2(:)'; y1 =y2(:)';   %生成测试样本数据。
figure; z1 =sim(net,[x1; y1]);   %求出测试样本数据的拟合输出值。
z2 = reshape(z1,size(x2)); surf(x2,y2,z2)   %画出拟合曲面。
```



## 三、实验内容

1. 神经网络对一元函数进行曲线拟合。
2. 神经网络对二元函数进行曲面拟合。
3. 神经网络在数据拟合具体实例上的应用。



# 四、实验报告

## 神经网络在数据拟合中的应用

实验名称：＿＿＿＿＿＿＿＿　　实验日期：＿＿＿年＿＿月＿＿日
姓名：＿＿＿＿＿＿＿＿　　班级学号：＿＿＿＿＿＿＿＿
成绩：＿＿＿＿＿＿＿＿

### （一）实验内容及步骤

1. 已知数据，见表 3－12－2。

**表 3－12－2**

| x | 0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| y | 0.3 | 0.5 | 1 | 1.4 | 1.6 | 1.9 | 0.6 | 0.4 | 0.8 | 1.5 | 2 |

利用神经网络在［0.1］区间上绘制出样本对应的函数曲线。尝试不同的神经网络结构和训练算法，将神经网络的曲线拟合结果和实验十一的三次多项式拟合结果进行比较。

（1）利用神经网络算法拟合曲线。

程序：

运行结果：

（2）不同神经网络结构和训练算法的比较。

（3）与三次多项式拟合进行比较。

2. “人口问题”是我国最大社会问题之一，估计人口数量和发展趋势是我们制定一系列相关政策的基础。有人口统计年鉴，可查的我国从 1949 ~ 1994 年人口数据资料见表 3 - 12 - 3。

**表 3 - 12 - 3**

| 年份 | 1949 | 1954 | 1959 | 1964 | 1969 | 1974 | 1979 | 1984 | 1989 | 1994 |
|---|---|---|---|---|---|---|---|---|---|---|
| 人口数（百万） | 541.67 | 602.66 | 672.09 | 704.99 | 806.71 | 908.59 | 975.42 | 1 034.75 | 1 106.76 | 1 176.74 |

分析：

（1）在直角坐标系上作出人口数的图像。

（2）用神经网络方法拟合数据，并算出 1999 年人口数。

程序：

运行结果：

3. 已知某山区地形选点测量坐标数据为：

x = 0　0.5　1　1.5　2　2.5　3　3.5　4　4.5　5

y = 0　0.5　1　1.5　2　2.5　3　3.5　4　4.5　5　5.5　6

海拔高度数据为：

z = 89 90 87 85 92 91 96 93 90 87 82
　92 96 98 99 95 91 89 86 84 82 84
　96 98 95 92 90 88 85 84 83 81 85
　80 81 82 89 95 96 93 92 89 86 86
　82 85 87 98 99 96 97 88 85 82 83
　82 85 89 94 95 93 92 91 86 84 88
　88 92 93 94 95 89 87 86 83 81 92
　92 96 97 98 96 93 95 84 82 81 84
　85 85 81 82 80 80 81 85 90 93 95



84 86 81 98 99 98 97 96 95 84 87
80 81 85 82 83 84 87 90 95 86 88
80 82 81 84 85 86 83 82 81 80 82
87 88 89 98 99 97 96 98 94 92 87

实验十一的第 2 题中已画出其原始地貌图，利用神经网络画出其加密后的地貌图。加密后横坐标数据为 50 个，纵坐标数据为 60 个。

（1）程序：

运行结果：

（2）拟合效果如何？更改神经网络结构能否改善拟合效果？产生此结果的原因是什么？

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

1. 将第 3 题的结果与用插值法的结果进行比较，哪种效果好？为什么？

2. 实验小结。



实验十三

# 最佳营销策略
（综合性实验）

## 一、实验目的

综合应用 MATLAB 解决实际问题。

## 二、实验原理

函数极值

MATLAB 提供了基于单纯形算法求解函数极值的函数 fminunc，它用于求函数的最小值，其调用格式为：

```
[x,fval]= fminunc('fun',x0,option)
```

其中 x 为返回的解，fval 是解向量处对应的函数值 f（x）向量。fun 是用于定义需求解的目标函数文件名，x0 是求解过程的初值。

MATLAB 没有专门提供求函数最大值的函数，但只要注意到 -f（x）在区间（a，b）上的最小值就是 f（x）在（a，b）的最大值。

例如：求 f（x）=x3 -2x -5 在［0，5］内的最小值点。

（1）建立函数文件 mymin. m。

```
function fx = mymin(x)
fx = x.^3 - 2 * x - 5;
```

（2）调用 fmin 函数求最小值点。

```
x = fminunc('mymin',0)
x =
    0.8165
```

## 三、实验内容

某公司有一批以每桶2元购进的彩漆为了获得较高的利润希望以较高的价格卖出，但价格越高，售出量就越少，二者之间的关系由表3－13－1给出。于是打算增加广告投入来促销。而广告费与销售量的关系可由销售增长因子来描述。例如，投入3万元的广告费，销售因子为1.85，意味着做广告后的销售量将是未做广告销售量的1.85倍。根据经验，广告费与销售因子的关系如表3－13－2所示，现请你作出决策：投入多少广告费和售价为多少时所获得的利润最大?

**表3－13－1**

| 售　价 | 2.00 | 2.50 | 3.00 | 3.50 | 4.00 | 4.50 | 5.00 | 5.50 | 6.00 |
|---|---|---|---|---|---|---|---|---|---|
| 预期销售量（千桶） | 41 | 38 | 34 | 32 | 29 | 28 | 25 | 22 | 20 |

**表3－13－2**

| 广告费（千元） | 0 | 10 | 20 | 30 | 40 | 50 | 60 | 70 |
|---|---|---|---|---|---|---|---|---|
| 销售增长因子 | 1.00 | 1.40 | 1.70 | 1.85 | 1.95 | 2.00 | 1.95 | 1.80 |



# 四、实验报告

## 最佳营销策略

实验名称：______________________ 实验日期：________年____月____日

姓名：__________________________ 班级学号：______________________

成绩：__________________________

### （一）实验内容及步骤

1. 用描点法画出预期销售量—售价；销售增长因子—广告费的关系图。

程序：

运行结果：

2. 建立上述两个函数关系。（提示：可设计一个自己认为合理的方法，如可以用曲线拟合的方法。）

程序：

运行结果：



3. 设 x 为售价，y 为广告费，P 为所得利润，建立 P 关于 x 和 y 的函数关系。

4. 用多元函数的极值理论计算并回答此问题。

程序：

运行结果：

## （二）实验中出现的问题及实验结果分析

## （三）实验小结及思考

# 实验十四

# 放射性废物的处理问题

## （综合性实验）

## 一、实验目的

综合应用 MATLAB 解决实际问题。

## 二、实验内容

美国原子能委员会几年来一直用这样的方法来处理放射性废料：把废料装入密封的圆桶里，然后将圆桶扔入深度为 91.14m 的海里。生态学家和工程师反对这种做法。因为通过大量的实验证明：如果当圆桶到达海底时速度超过 12.2m/s，圆桶会因撞击海底而破裂。但美国原子能委员会坚持认为圆桶到达海底时不会超过这个速度。现假设圆桶所受的重力为 $W=239.46g$（N），海水作用于圆桶的浮力为 $B=213.528g$（N），圆桶下沉时所阻力与下沉速度成正比，比例系数为 $C=0.19$kgs/m，并设圆桶刚被抛入海里时，它的速度为零。现在问题的焦点集中于圆桶到达海底时，其速度是否超过 12.2m/s?





# 三、实验报告

## 放射性废物的处理问题

实验名称：____________________　　实验日期：_______年____月____日
姓名：______________________　　班级学号：____________________
成绩：______________________

## （一）实验内容及步骤

1. 建立方程模型并求解（应是一个关于速度的一阶线性非齐次方程）。
模型：

求出圆桶重力（G）、浮力（F）、比例系数的值。将其代入到模型当中。并求解该模型。求解：

2. 你得到的解是否为：

$$V(t)=\frac{W-B}{C}(1-e^{-\frac{cg}{w}t}) \tag{3-14-1}$$

在式（3－14－1）的两边对t积分，得圆桶的位置与t的函数关系。请你写出这个关系。

3. 原子能委员会提出一简便的估算方法该方法为：当 $t \to +\infty$ 时，圆桶到达海底时的速度 $V(t)$ 趋于极限 $Vt$，显然，$V(t) \leqslant Vt$（为什么）。如果 $Vt < 12.2\text{m/s}$，那么结果将有利于美国原子能委员会。请你计算一下 $Vt$ 等于多少？

4. 若步骤 3 不能解决问题，继续寻找解题方法。

已知圆桶的位置 x 也是时间 t 的函数，因此，我们也可以把速度 v 看成是位置的函数。于是有 $V(t) = v(x(t))$。且当 $t = 0$ 时，$x(0) = 0$，$v(0) = V(0) = 0$。根据链式微分法则，可以得到新的模型：

$$\frac{W}{g} v \frac{dv}{dx} = W - B - Cv,\ v(0) = 0 \tag{3-14-2}$$

在式（3-14-2）中，令 $C = 0$，（想一想，这意味着什么？）得式（3-14-2）的近似模型：

$$\frac{W}{g} u \frac{du}{dx} = W - B,\ u(0) = 0 \tag{3-14-3}$$

注意：此处 u 是 v 的近似解。

求解方程式（3-14-3）

得到速度 v 与位置 x 的函数关系式为：

计算当 $x = 91.14$，$v(91.14)$ 的近似值：

$u(91.14)$ = ____________________

根据以上结果，你的结论是什么？

5. 利用微分方程数值解的方法求解该问题，并将其与解析解比较。

程序：



### （二）实验中出现的问题及实验结果分析

### （三）实验小结及思考





# 附录一

# MATLAB 指令索引

MATLAB 关键符（词）的说明

A MATLAB 的标点及符号

1. 算术运算符 Arithmetic operators

+：加；正号。

-：减；负号。

*：矩阵乘。

.*：数组乘。

\：矩阵左除。

/：矩阵右除。

.\：数组左除。

./：数组右除。

^：矩阵幂。

.^：数组幂。

2. 关系运算符 Relational operators

==：等于。

~=：不等于。

<：小于。

>：大于。

<=：小于等于。

>=：大于等于。

3. 逻辑运算符 Logical operators

&：逻辑与。

|：逻辑或。

~：逻辑非。

4. 特殊符号 Special characters

,：逗号。

;：分号。

 ：空格。

.：小数点号、构架域号。

::冒号。

...：续行号。

''：单引号。

 ：共轭转置号。

.'：转置号。

=：赋值号。

_ ：下连符。

!：调用 DOS 操作指令号。

( )：圆括号。

[ ]：方括号。

[ ]：空阵。

{ }：花括号。

%：注释号。

%#：编译注记。

A a

abs：绝对值、模、字符的 ASCII 码值。

acos：反余弦。

acosh：反双曲余弦。

acot：反余切。

acoth：反双曲余切。

acsc：反余割。

acsch：反双曲余割。

align：启动图形对象几何位置排列工具。

all：所有元素非零为真。

angle：相角。

ans：表达式计算结果的缺省变量名。

any：所有元素非全零为真。

area：面域图。

argnames：函数 M 文件宗量名。
asec：反正割。
asech：反双曲正割。
asin：反正弦。
asinh：反双曲正弦。
assignin：向变量赋值。
atan：反正切。
atan2：四象限反正切。
atanh：反双曲正切。
autumn：红黄调秋色图阵。
axes：创建轴对象的低层指令。
axis：控制轴刻度和风格的高层指令。
B b
bar：二维直方图。
bar3：三维直方图。
bar3h：三维水平直方图。
barh：二维水平直方图。
base2dec X：进制转换为十进制。
bin2dec：二进制转换为十进制。
blanks：创建空格串。
bone：蓝色调黑白色图阵。
box：框状坐标轴。
break while 或 for：环中断指令。
brighten：亮度控制。
C c
capture：（3 版以前）捕获当前图形。
cart2pol：直角坐标变为极或柱坐标。
cart2sph：直角坐标变为球坐标。
cat：串接成高维数组。
caxis：色标尺刻度。
cd：指定当前目录。
cdedit：启动用户菜单、控件回调函数设计工具。
cdf2rdf：复数特征值对角阵转为实数块对角阵。
ceil：向正无穷取整。
cell：创建元胞数组。
cell2struct：元胞数组转换为构架数组。
celldisp：显示元胞数组内容。
cellplot：元胞数组内部结构图示。
char：把数值、符号、内联类转换为字符对象。

chi2cdf：分布累计概率函数。

chi2inv：分布逆累计概率函数。

chi2pdf：分布概率密度函数。

chi2rnd：分布随机数发生器。

chol Cholesky：分解。

clabel：等位线标识。

cla：清除当前轴。

class：获知对象类别或创建对象。

clc：清除指令窗。

clear：清除内存变量和函数。

clf：清除图对象。

clock：时钟。

colorcube：三浓淡多彩交叉色图矩阵。

colordef：设置色彩缺省值。

colormap：色图。

colspace：列空间的基。

close：关闭指定窗口。

colperm：列排序置换向量。

comet：彗星状轨迹图。

comet3：三维彗星轨迹图。

compass：射线图。

compose：求复合函数。

cond：（逆）条件数。

condeig：计算特征值、特征向量同时给出条件数。

condest：范 -1 条件数估计值。

conj：复数共轭。

contour：等位线。

contourf：填色等位线。

contour3：三维等位线。

contourslice：四维切片等位线图。

conv：多项式乘、卷积。

cool：青紫调冷色图。

copper：古铜调色图。

cos：余弦。

cosh：双曲余弦。

cot：余切。

coth：双曲余切。

cplxpair：复数共轭成对排列。

csc：余割。

csch：双曲余割。
cumsum：元素累计和。
cumtrapz：累计梯形积分。
cylinder：创建圆柱。
D d
dblquad：二重数值积分。
deal：分配宗量。
deblank：删去串尾部的空格符。
dec2base：十进制转换为 X 进制。
dec2bin：十进制转换为二进制。
dec2hex：十进制转换为十六进制。
deconv：多项式除、解卷。
delaunay Delaunay。三角剖分。
del2：离散 Laplacian 差分。
demo MATLAB：演示：
det：行列式。
diag：矩阵对角元素提取、创建对角阵。
diary MATLAB：指令窗文本内容记录。
diff：数值差分、符号微分。
digits：符号计算中设置符号数值的精度。
dir：目录列表。
disp：显示数组。
display：显示对象内容的重载函数。
dlinmod：离散系统的线性化模型。
dmperm：矩阵 Dulmage-Mendelsohn 分解。
dos：执行 DOS 指令并返回结果。
double：把其他类型对象转换为双精度数值。
drawnow：更新事件队列强迫 MATLAB 刷新屏幕。
dsolve：符号计算解微分方程。
E e
echo：M 文件被执行指令的显示。
edit：启动 M 文件编辑器。
eig：求特征值和特征向量。
eigs：求指定的几个特征值。
end：控制流 FOR 等结构体的结尾元素下标。
eps：浮点相对精度。
error：显示出错信息并中断执行。
errortrap：错误发生后程序是否继续执行的控制。
erf：误差函数。

erfc：误差补函数。
erfcx：刻度误差补函数。
erfinv：逆误差函数。
errorbar：带误差限的曲线图。
etreeplot：画消去树。
eval：串演算指令。
evalin：跨空间串演算指令。
exist：检查变量或函数是否已定义。
exit：退出 MATLAB 环境。
exp：指数函数。
expand：符号计算中的展开操作。
expint：指数积分函数。
expm：常用矩阵指数函数。
expm1：Pade 法求矩阵指数。
expm2：Taylor 法求矩阵指数。
expm3：特征值分解法求矩阵指数。
eye：单位阵。
ezcontour：画等位线的简捷指令。
ezcontourf：画填色等位线的简捷指令。
ezgraph3：画表面图的通用简捷指令。
ezmesh：画网线图的简捷指令。
ezmeshc：画带等位线的网线图的简捷指令。
ezplot：画二维曲线的简捷指令。
ezplot3：画三维曲线的简捷指令。
ezpolar：画极坐标图的简捷指令。
ezsurf：画表面图的简捷指令。
ezsurfc：画带等位线的表面图的简捷指令。
F f
factor：符号计算的因式分解。
feather：羽毛图。
feedback：反馈连接。
feval：执行由串指定的函数。
fft：离散 Fourier 变换。
fft2：二维离散 Fourier 变换。
fftn：高维离散 Fourier 变换。
fftshift：直流分量对中的谱。
fieldnames：构架域名。
figure：创建图形窗。
fill3：三维多边形填色图。

find：寻找非零元素下标。
findobj：寻找具有指定属性的对象图柄。
findstr：寻找短串的起始字符下标。
findsym：机器确定内存中的符号变量。
finverse：符号计算中求反函数。
fix：向零取整。
flag：红白蓝黑交错色图阵。
fliplr：矩阵的左右翻转。
flipud：矩阵的上下翻转。
flipdim：矩阵沿指定维翻转。
floor：向负无穷取整。
flops：浮点运算次数。
flow：MATLAB 提供的演示数据。
fmin：求单变量非线性函数极小值点（旧版）。
fminbnd：求单变量非线性函数极小值点。
fmins：单纯形算法求多变量函数极小值点（旧版）。
fminunc：拟牛顿法求多变量函数极小值点。
fminsearch：单纯形算法求多变量函数极小值点。
fnder：对样条函数求导。
fnint：利用样条函数求积分。
fnval：计算样条函数区间内任意一点的值。
fnplt：绘制样条函数图形。
fopen：打开外部文件。
for：构成 for 环用。
format：设置输出格式。
fourier：Fourier 变换。
fplot：返函绘图指令。
fprintf：设置显示格式。
fread：从文件读二进制数据。
fsolve：求多元函数的零点。
full：把稀疏矩阵转换为非稀疏矩阵。
funm：计算一般矩阵函数。
funtool：函数计算器图形用户界面。
fzero：求单变量非线性函数的零点。
G g
gamma：函数。
gammainc：不完全函数。
gammaln：函数的对数。
gca：获得当前轴句柄。

gcbo：获得正执行“回调”的对象句柄。
gcf：获得当前图对象句柄。
gco：获得当前对象句柄。
geomean：几何平均值。
get：获知对象属性。
getfield：获知构架数组的域。
getframe：获取影片的帧画面。
ginput：从图形窗获取数据。
global：定义全局变量。
gplot：依图论法则画图。
gradient：近似梯度。
gray：黑白灰度。
grid：画分格线。
griddata：规则化数据和曲面拟合。
gtext：由鼠标放置注释文字。
guide：启动图形用户界面交互设计工具。
H h
harmmean：调和平均值。
help：在线帮助。
helpwin：交互式在线帮助。
helpdesk：打开超文本形式用户指南。
hex2dec：十六进制转换为十进制。
hex2num：十六进制转换为浮点数。
hidden：透视和消隐开关。
hilb：Hilbert 矩阵。
hist：频数计算或频数直方图。
histc：端点定位频数直方图。
histfit：带正态拟合的频数直方图。
hold：当前图上重画的切换开关。
horner：分解成嵌套形式。
hot：黑红黄白色图。
hsv：饱和色图。
I i
if-else-elseif：条件分支结构。
ifft：离散 Fourier 反变换。
ifft2：二维离散 Fourier 反变换。
ifftn：高维离散 Fourier 反变换。
ifftshift：直流分量对中的谱的反操作。
ifourier：Fourier 反变换。

i，j：缺省的“虚单元”变量。
ilaplace：Laplace 反变换。
imag：复数虚部。
image：显示图像。
imagesc：显示亮度图像。
imfinfo：获取图形文件信息。
imread：从文件读取图像。
imwrite：把图像写成文件。
ind2sub：单下标转变为多下标。
inf：无穷大。
info：MathWorks 公司网点地址。
inline：构造内联函数对象。
inmem：列出内存中的函数名。
input：提示用户输入。
inputname：输入宗量名。
int：符号积分。
int2str：把整数数组转换为串数组。
interp1：一维插值。
interp2：二维插值。
interp3：三维插值。
interpn：N 维插值。
interpft：利用 FFT 插值。
intro：MATLAB 自带的入门引导。
inv：求矩阵逆。
invhilb：Hilbert 矩阵的准确逆。
ipermute：广义反转置。
isa：检测是否给定类的对象。
ischar：若是字符串则为真。
isequal：若两数组相同则为真。
isempty：若是空阵则为真。
isfinite：若全部元素都有限则为真。
isfield：若是构架域则为真。
isglobal：若是全局变量则为真。
ishandle：若是图形句柄则为真。
ishold：若当前图形处于保留状态则为真。
isieee：若计算机执行 IEEE 规则则为真。
isinf：若是无穷数据则为真。
isletter：若是英文字母则为真。
islogical：若是逻辑数组则为真。

ismember：检查是否属于指定集。
isnan：若是非数则为真。
isnumeric：若是数值数组则为真。
isobject：若是对象则为真。
isprime：若是质数则为真。
isreal：若是实数则为真。
isspace：若是空格则为真。
issparse：若是稀疏矩阵则为真。
isstruct：若是构架则为真。
isstudent：若是 MATLAB 学生版则为真。
iztrans：符号计算 Z 反变换。
J j ， K k
jacobian：符号计算中求 Jacobian 矩阵。
jet：蓝头红尾饱和色。
jordan：符号计算中获得 Jordan 标准型。
keyboard：键盘获得控制权。
kron：Kronecker 乘法规则产生的数组。
L l
laplace：Laplace 变换。
lasterr：显示最新出错信息。
lastwarn：显示最新警告信息。
leastsq：解非线性最小二乘问题（旧版）。
legend：图形图例。
lighting：照明模式。
line：创建线对象。
lines：采用 plot 画线色。
linmod：获连续系统的线性化模型。
linmod2：获连续系统的线性化精良模型。
linspace：线性等分向量。
ln：矩阵自然对数。
load：从 MAT 文件读取变量。
log：自然对数。
log10：常用对数。
log2：底为 2 的对数。
loglog：双对数刻度图形。
logm：矩阵对数。
logspace：对数分度向量。
lookfor：按关键字搜索 M 文件。
lower：转换为小写字母。

lsqnonlin：解非线性最小二乘问题。
lu：LU 分解。
M m
mad：平均绝对值偏差。
magic：魔方阵。
maple & nb，sp：运作 Maple 格式指令。
mat2str：把数值数组转换成输入形态串数组。
material：材料反射模式。
max：找向量中最大元素。
mbuild：产生 EXE 文件编译环境的预设置指令。
mcc：创建 MEX 或 EXE 文件的编译指令。
mean：求向量元素的平均值。
median：求中位数。
menuedit：启动设计用户菜单的交互式编辑工具。
mesh：网线图。
meshz：垂帘网线图。
meshgrid：产生“格点”矩阵。
methods：获知对指定类定义的所有方法函数。
mex：产生 MEX 文件编译环境的预设置指令。
mfunlis：能被 mfun 计算的 MAPLE 经典函数列表。
mhelp：引出 Maple 的在线帮助。
min：找向量中最小元素。
mkdir：创建目录。
mkpp：逐段多项式数据的明晰化。
mod：模运算。
more：指令窗中内容的分页显示。
movie：放映影片动画。
moviein：影片帧画面的内存预置。
mtaylor：符号计算多变量 Taylor 级数展开。
N n
ndims：求数组维数。
NaN：非数（预定义）变量。
nargchk：输入宗量数验证。
nargin：函数输入宗量数。
nargout：函数输出宗量数。
ndgrid：产生高维格点矩阵。
newplot：准备新的缺省图、轴。
nextpow2：取最接近的较大 2 次幂。
nnz：矩阵的非零元素总数。

nonzeros：矩阵的非零元素。
norm：矩阵或向量范数。
normcdf：正态分布累计概率密度函数。
normest：估计矩阵 2 范数。
norminv：正态分布逆累计概率密度函数。
normpdf：正态分布概率密度函数。
normrnd：正态随机数发生器。
notebook：启动 MATLAB 和 Word 的集成环境。
null：零空间。
num2str：把非整数数组转换为串。
numden：获取最小公分母和相应的分子表达式。
nzmax：指定存放非零元素所需内存。
O o
ode1：非 Stiff 微分方程变步长解算器。
ode15s：Stiff 微分方程变步长解算器。
ode23t：适度 Stiff 微分方程解算器。
ode23tb：Stiff 微分方程解算器。
ode45：非 Stiff 微分方程变步长解算器。
odefile：ODE 文件模板。
odeget：获知 ODE 选项设置参数。
odephas2 ODE：输出函数的二维相平面图。
odephas3 ODE：输出函数的三维相空间图。
odeplot ODE：输出函数的时间轨迹图。
odeprint：在 MATLAB 指令窗显示结果。
odeset：创建或改写 ODE 选项构架参数值。
ones：全 1 数组。
optimset：创建或改写优化泛函指令的选项参数值。
orient：设定图形的排放方式。
orth：值空间正交化。
P p
pack：收集 MATLAB 内存碎块扩大内存。
pagedlg：调出图形排版对话框。
patch：创建块对象。
path：设置 MATLAB 搜索路径的指令。
pathtool：搜索路径管理器。
pause：暂停。
pcode：创建预解译 P 码文件。
pcolor：伪彩图。
peaks：MATLAB 提供的典型三维曲面。

permute：广义转置。
pi：（预定义变量）圆周率。
pie：二维饼图。
pie3：三维饼图。
pink：粉红色图矩阵。
pinv：伪逆。
plot：平面线图。
plot3：三维线图。
plotmatrix：矩阵的散点图。
plotyy：双纵坐标图。
poissinv：泊松分布逆累计概率分布函数。
poissrnd：泊松分布随机数发生器。
pol2cart：极或柱坐标变为直角坐标。
polar：极坐标图。
poly：矩阵的特征多项式、根集对应的多项式。
poly2str：以习惯方式显示多项式。
poly2sym：双精度多项式系数转变为向量符号多项式。
polyder：多项式导数。
polyfit：数据的多项式拟合。
polyval：计算多项式的值。
polyvalm：计算矩阵多项式。
pow2：2 的幂。
ppval：计算分段多项式。
pretty：以习惯方式显示符号表达式。
print：打印图形或 SIMULINK 模型。
printsys：以习惯方式显示有理分式。
prism：光谱色图矩阵。
procread：向 MAPLE 输送计算程序。
profile：函数文件性能评估器。
propedit：图形对象属性编辑器。
pwd：显示当前工作目录。
Q q
quad：低阶法计算数值积分。
quad8：高阶法计算数值积分。
quit：推出 MATLAB 环境。
quiver：二维方向箭头图。
quiver3：三维方向箭头图。
R r
rand：产生均匀分布随机数。

randn：产生正态分布随机数。
randperm：随机置换向量。
range：样本极差。
rank：矩阵的秩。
rats：有理输出。
rcond：矩阵倒条件数估计。
real：复数的实部。
reallog：在实数域内计算自然对数。
realpow：在实数域内计算乘方。
realsqrt：在实数域内计算平方根。
realmax：最大正浮点数。
realmin：最小正浮点数。
rectangle：画长方框。
rem：求余数。
repmat：铺放模块数组。
reshape：改变数组维数、大小。
residue：部分分式展开。
return：返回。
ribbon：把二维曲线画成三维彩带图。
rmfield：删去构架的域。
roots：求多项式的根。
rose：数扇形图。
rot90：矩阵旋转 90 度。
rotate：指定的原点和方向旋转。
rotate3d：启动三维图形视角的交互设置功能。
round：向最近整数圆整。
rref：简化矩阵为梯形形式。
rsf2csf：实数块对角阵转为复数特征值对角阵。
rsums：Riemann 的和。
S s
save：把内存变量保存为文件。
scatter：散点图。
scatter3：三维散点图。
sec：正割。
sech：双曲正割。
semilogx：X 轴对数刻度坐标图。
semilogy：Y 轴对数刻度坐标图。
series：串联连接。
set：设置图形对象属性。

setfield：设置构架数组的域。
setstr：将 ASCII 码转换为字符的旧版指令。
sign：根据符号取值函数。
signum：符号计算中的符号取值函数。
sim：运行 SIMULINK 模型。
simget：获取 SIMULINK 模型设置的仿真参数。
simple：寻找最短形式的符号解。
simplify：符号计算中进行简化操作。
simset：对 SIMULINK 模型的仿真参数进行设置。
simulink：启动 SIMULINK 模块库浏览器。
sin：正弦。
sinh：双曲正弦。
size：矩阵的大小。
slice：立体切片图。
solve：求代数方程的符号解。
spalloc：为非零元素配置内存。
sparse：创建稀疏矩阵。
spconvert：把外部数据转换为稀疏矩阵。
spdiags：稀疏对角阵。
spfun：求非零元素的函数值。
sph2cart：球坐标变为直角坐标。
sphere：产生球面。
spinmap：色图彩色的周期变化。
spline：样条插值。
spones：用 1 置换非零元素。
sprandsym：稀疏随机对称阵。
sprank：结构秩。
spring：紫黄调春色图。
sprintf：把格式数据写成串。
spy：画稀疏结构图。
sqrt：平方根。
sqrtm：方根矩阵。
squeeze：删去大小为 1 的“孤维”。
sscanf：按指定格式读串。
stairs：阶梯图。
std：标准差。
stem：二维杆图。
step：阶跃响应指令。
str2double：串转换为双精度值。

str2mat：创建多行串数组。
str2num：串转换为数。
strcat：接成长串。
strcmp：串比较。
strjust：串对齐。
strmatch：搜索指定串。
strncmp：串中前若干字符比较。
strrep：串替换。
strtok：寻找第一间隔符前的内容。
struct：创建构架数组。
struct2cell：把构架转换为元胞数组。
strvcat：创建多行串数组。
sub2ind：多下标转换为单下标。
subexpr：通过子表达式重写符号对象。
subplot：创建子图。
subs：符号计算中的符号变量置换。
subspace：两子空间夹角。
sum：元素和。
summer：绿黄调夏色图。
superiorto：设定优先级。
surf：三维着色表面图。
surface：创建面对象。
surfc：带等位线的表面图。
surfl：带光照的三维表面图。
surfnorm：空间表面的法线。
svd：奇异值分解。
svds：求指定的若干奇异值。
switch-case-otherwise：多分支结构。
sym2poly：符号多项式转变为双精度多项式系数向量。
symmmd：对称最小度排序。
symrcm：反向 Cuthill-McKee 排序。
syms：创建多个符号对象。
T t
tan：正切。
tanh：双曲正切。
taylortool：进行 Taylor 逼近分析的交互界面。
text：文字注释。
tf：创建传递函数对象。
tic：启动计时器。



title：图名。
toc：关闭计时器。
trapz：梯形法数值积分。
treelayout：展开树林。
treeplot：画树图。
tril：下三角阵。
trim：求系统平衡点。
trimesh：不规则格点网线图。
trisurf：不规则格点表面图。
triu：上三角阵。
try-catch：控制流中的 Try-catch 结构。
type：显示 M 文件。
U u
uicontextmenu：创建现场菜单。
uicontrol：创建用户控件。
uimenu：创建用户菜单。
unmkpp：逐段多项式数据的反明晰化。
unwrap：自然态相角。
upper：转换为大写字母。
V v
var：方差。
varargin：变长度输入宗量。
varargout：变长度输出宗量。
vectorize：使串表达式或内联函数适于数组运算。
ver：版本信息的获取 view 三维图形的视角控制。
voronoi Voronoi：多边形。
vpa：任意精度（符号类）数值。
W w
warning：显示警告信息。
what：列出当前目录上的文件。
whatsnew：显示 MATLAB 中 Readme 文件的内容。
which：确定函数、文件的位置。
while：控制流中的 While 环结构。
white：全白色图矩阵。
whitebg：指定轴的背景色。
who：列出内存中的变量名。
whos：列出内存中变量的详细信息。
winter：蓝绿调冬色图。
workspace：启动内存浏览器。



X x ， Y y ， Z z

xlabel X：轴名。

xor：或非逻辑。

yesinput：智能输入指令。

ylabel Y：轴名。

zeros：全零数组。

zlabel Z：轴名。

zoom：图形的变焦放大和缩小。

ztrans：符号计算 Z 变换。

# 附录二

# 部分实验内容参考程序

## 实验一

2. (1) 5. 54^2 + cos(3 * pi/5) * sqrt(17. 89)/3. 5-exp(2)

   (3)22. 9291

3. (1)A = [4 2 8;2 5 6;3 1 9]　　B = [1 1 1;2 2 2;3 3 3]

   (2) A(2,3) = 6　A(6) = 1　$A(:,2) = \begin{pmatrix} 2 \\ 5 \\ 1 \end{pmatrix}$　A(3,:) = (3　1　9)

$$A(:,1:2:3) = \begin{pmatrix} 4 & 8 \\ 2 & 6 \\ 3 & 9 \end{pmatrix} \qquad A(:,3).*B(:,2) = \begin{pmatrix} 8 \\ 12 \\ 27 \end{pmatrix}$$

$$A(:,3)*B(2,:) = \begin{pmatrix} 12 & 12 & 12 \\ 16 & 16 & 16 \\ 18 & 18 & 18 \end{pmatrix}$$

$$A*B = \begin{pmatrix} 32 & 32 & 32 \\ 30 & 30 & 30 \\ 32 & 32 & 32 \end{pmatrix} \qquad A.*B = \begin{pmatrix} 4 & 2 & 8 \\ 4 & 10 & 12 \\ 9 & 3 & 27 \end{pmatrix} \qquad A\hat{}2 = \begin{pmatrix} 44 & 26 & 116 \\ 36 & 35 & 100 \\ 41 & 20 & 111 \end{pmatrix}$$



$$A.\hat{}2=\begin{pmatrix}16 & 4 & 64\\ 4 & 25 & 36\\ 9 & 1 & 81\end{pmatrix}\qquad B/A=\begin{pmatrix}0.5 & 0.0769 & -0.3846\\ 1 & 0.1538 & -0.7692\\ 1.5 & 0.2308 & -1.1538\end{pmatrix}$$

$$B./A=\begin{pmatrix}0.25 & 0.5 & 0.125\\ 1 & 0.4 & 0.333\\ 1 & 3 & 0.333\end{pmatrix}\qquad B.\backslash A=\begin{pmatrix}4 & 2 & 8\\ 1 & 2.5 & 3\\ 1 & 0.333 & 3\end{pmatrix}$$

4. (1)清除 MATLAB 工作空间中保存的变量　　清除指令窗中显示内容
   (2)M = magic(5)
   (3)a = M(3,4)
   (4)b = M(2:4,[2,5])
5. (1)2010 * eye(10)
   (2)6 * (ones(8) - eye(8))
6. (1)E = eye(3)
   (2)R = rand(3,2)
   (3)O = zeros(2,3)
   (4)D = diag([2,1/2])　% 对角线元素的具体数值可任取,此例中取 2 和 1/2
   (5)A = [E R;O D]

## 实验二

4. (1)

```
t = 0:pi/10:2 * pi;
x = sin(t);
y = cos(t);
z = sin(2 * t);
plot(t,x,'--k * ',t,y,'-rs',t,z,':bo')
```

(4)在 figure 窗口中选择"edit"菜单下的"copy figure"选项,再在 word 文档中粘贴即可。

5. (1)

```
t = 0:pi/10:10 * pi;
x = 2 * (cos(t) + t. * sin(t));
y = 2 * (sin(t) - t. * cos(t));
z = 1.5 * t;
plot3(x,y,z,':b','linewidth',3);
```

(2)

```
xlabel('x');
ylabel('y');
zlabel('z');
title('三维螺旋线');
```

6. (1)

```
x = -1.5 * pi:pi/21:1.5 * pi;
```



```
y = -1.5 * pi:pi/21:1.5 * pi;
z = cos(x). * sin(y). /y;
[X, Y] = meshgrid(x, y);
Z = cos(X). * sin(Y). /Y;
(2) subplot(2, 2, 1), plot3(x, y, z, 'linewidth', 2);title('三维曲线图');
subplot(2, 2, 2), mesh(X, Y, Z);title('网格图');
subplot(2, 2, 3), surf(X, Y, Z);title('光滑曲面图');shading interp;
(3)
i = find(Z < -0.1);
Z(i) = nan;
subplot(2, 2,4), surf(X, Y, Z);title('镂空图');shading interp;
```

## 实验三

4. (1)

```
function y = ff(x)
if x > = -1&x <0
        y = x + 1;
end
if   x > =0&x <1
        y = 1;
end
if   x > =1&x < =2
        y = x^2;
end
```

5. (1)

```
while 1
score = input('Please input score:');
if score <0 break;
else
switch fix(score/10)
     case {9, 10}
               result = 'A';
     case 8
               result = 'B';
     case 7
               result = 'C';
     case 6
               result = 'D';
     otherwise
```



```
                  result = 'E';
end
disp(['The result is',result])
end
end
```

6. (1) function y = jc(n)

```
m = 1;
for i = 1:n
      m = m * i;
end
fprintf('The result is %g\n',m)
```

7. (1)

```
s = 0;i = 0;
while(s < = 1000)
      i = i + 2;
      s = s + i;
end
fprintf('s = %d,n = %d\n',s - i,(i - 2)/2)
```

8. (1)解法一:

```
x = [ ];
for t = 100:999
      i = fix(t/100);
      j = fix((t - i * 100)/10);
      k = t - i * 100 - j * 10;
      if t == i^3 + j^3 + k^3
            x = [x t];
      end
end
disp(x)
```

解法二:

```
x = [ ];
for i = 1:9
   for j = 0:9
      for k = 0:9
              if i^3 + j^3 + k^3 == 100 * i + 10 * j + k
                    t = 100 * i + 10 * j + k;
                    x = [x t];
              end
         end
```



```
   end
end
      disp(x)
```

## 实验四

```
5. (1)syms x;
f=(1/x^3+6/x^2+12/x+8)^(1/3);
f=simple(f);
6. a.(1) syms x;
f=(cos(x)-exp(-x^2/2))/x^4;
limit(f)
b.(1) syms x;
f='(1+2*t/x)^(3*x)';
limit(f,x,+inf)
7. (1)syms x;
y=exp(2*x)*log(x^2+1)*tan(-x);
y1=diff(y)
y2=diff(y,x,2)
(2)syms x y;
z=(x^2+y^2)*exp((x^2+y^2)/(x*y));
z1=diff(z,x)
z2=diff(z,x,2)
z3=diff(z1,y)
8. (1)syms x ;
f=x^3*exp(-x^2);
f1=int(f)
(2)syms x;
f=x/(sin(x))^2;
f1=int(f,x,pi/4,pi/3)
(3)syms x y;
f=x*sin(x);
f1=int(f,x,y,sqrt(y))
f2=int(f1,y,0,1)
9. syms n;
f=1/((3*n-2)*(3*n+1));
s=symsum(f,n,1,+inf)
10. syms x;
f=sin(x)/(x^2+4*x+3);
f1=taylor(f,x,5)
```



```
f2 = taylor(f,x,7,2)
```

## 实验五

```
1. A = magic(4);
B = diag([1,2,3,4]);
C = B * A
5. a1 = [  1  -2   2  3]';
   a2 = [ -2   4  -1  3]';
   a3 = [ -1   2   0  3]';
   a4 = [  0   6   2  3]';
   a5 = [  2  -6   3  4]';
   A = [a1 a2 a3 a4 a5]
format rat      % 以有理格式输出
B = rref(A)     % 求 A 的行最简形
```

运行后的结果为

```
A =
     1     -2     -1     0     2
    -2      4      2     6    -6
     2     -1      0     2     3
     3      3      3     3     4
B =
     1      0     1/3    0    16/9
     0      1     2/3    0    -1/9
     0      0      0     1    -1/3
     0      0      0     0     0
```

从 B 中可以得到:向量 a1 a2 a4 为其中一个最大无关组

```
8. P = [3 0 4 -5 -7.3 5];
   X = roots(P)
   G = poly(X)
   X0 = [-1,2.4,5,6.8];
   f = polyval(P,X0)
9. f = [3,-5,2,-7,5,6];g = [3,5,-3];g1 = [0,0,0,g];
f + g1              % 求 f(x) + g(x)
f - g1              % 求 f(x) - g(x)
conv(f,g)           % 求 f(x) * g(x)
[Q,r] = deconv(f,g) % 求 f(x)/g(x)商式存入 Q,余式存入 r
11. (1)
format long;
x = 2:0.1:3;
```



```
y = 1. /log(x);
trapz(x,y)
```

(2) 
```
f = inline('1. /log(x)');
quad(f,2,3)
```

12. 
```
f = inline('exp( - x. ^2/2). * sin(x. ^2 + y)','x','y');
y = dblquad(f, -2,2, -1,1)
```

## 实验六

1. 
```
A = [1 -2 3 -4;0 1 -1 1;1 3 0 1;0 -7 1 1];
b = [4; -3;1; -3];
rank([A b])
ans =
    4
```

可知,方程组有唯一解.

```
X = A\b
X =
  -5. 000000000000000
   1. 500000000000000
   6. 000000000000000
   1. 500000000000000
```

2. (1)
```
A = [1 2 1 -1;3 6 -1 -3;5 10 1 -5];
rank(A)
ans =
     2
```

方程组有无穷多解.

3. (1)
```
A = [1 -2 3 -1 2;3 -1 5 -3 1;2 1 2 -2 -1] ;
b = [2 6 8]';
B = [A b];
n = 5;
R_A = rank(A)
R_B = rank(B)
```

4. 
```
A = [1 1 -3 -1;3 -1 -3 4;1 5 -9 -8];
b = [1 4 0]';
B = [A b];
n = 4;
R_A = rank(A)
R_B = rank(B)
format rat
if R_A = = R_B&R_A = = n
```



```
    X = A\b
  elseif R_A = = R_B&R_A < n
    X = A\b
    C = null(A,'r')
  else X = 'Equation has no solves'
end
```

5. solve('a * x^2 - b * x - 6 = 0')

6. [x,y] = solve('2 * x + 3 * y = 0','4 * x^2 + 3 * y = 1')

## 实验七

1. (1) syms x y

```
y = dsolve('Dy + y * tan(x) = cos(x)^2','x')
```

(2) diff(y,x) + y * tan(x) - cos(x)^2

```
simplify(diff(y,x) + y * tan(x) - cos(x)^2)
```

2. syms x y

```
y = dsolve('D2y = sin(2 * x) - y','y(0) = 0','Dy(0) = 1','x')
ezplot(y)
```

3. syms x y t

```
[x,y] = dsolve('Dx + x + y = 0','Dy + x - y = 0','x(0) = 1','y(0) = 0','t')
simple(x);
simple(y);
ezplot(x,y,[ -2,2]);
```

4. fun = inline('2 * y + x + 2','x','y');

```
[x,y] = ode23(fun,[0,1],1);
x';
y';
plot(x,y,'o - ')
```

5. (1) y = dsolve('Dy = 3 * y/x + x^3 * (exp(x) + cos(x)) - 2 * x','y(pi) = (exp(pi) + 2/pi) * pi^3','x')

(2) fun = inline('3 * y/x + x^3 * (exp(x) + cos(x)) - 2 * x','x','y');

```
[x,y] = ode45(fun,[pi,2 * pi],(exp(pi) + 2/pi) * pi^3);
```

(3) ezplot(y);hold on;

```
x';
y';
plot(x,y,'o - r');hold off;
```

6. (1) 编写函数文件 fun. m:

```
function ff = fun(x,y)
ff = [y(2);-2 * y(2) - y(1) + cos(x)];
```

(2) y0 = [0;3/2];



```
    [x,y] = ode45('fun',[0,10],y0);
    y1 = y(:,1);y2 = y(:,2);
plot(x,y1)
```

## 实验八

```
1. (1)ezplot('10 * x + exp(x) - 2'), grid on;
(2) f = inline('10 * x + exp(x) - 2');
    z = fzero(f,[0,0.5])
2. (1)function ff = fun1(x)
ff = [sin(x(1)) + x(2) + x(3)^2 * exp(x(1)) - 4;
      x(1) + x(2) * x(3);
      x(1) * x(2) * x(3)];
(2)[x,fval] = fsolve('fun1',[1,1,1])
3. (1)function ff = fun2(x)
ff = [3 * x(1) + 5 * x(2) + 6 * x(3);
      x(1) - 3 * x(2) - 6 * x(3) - 1;
      x(1)^2 + x(2)^2 + x(3)^2 - 9];
(2)[x,fval] = fsolve('fun2',[1,1,1])
    [x,fval] = fsolve('fun2',[-1,-1,-1])
5. (1)rate2.m
    function y = rate2(x)
    y = 500 * (1 + x)^(15 * 12) - 4.5 * ((1 + x)^(15 * 12) - 1)/x;   %计算月利率
    rate3.m
      function y = rate3(x)
    y = 500 * (1 + x)^20 - 45 * ((1 + x)^20 - 1)/x;      %计算年利率
(2)[x2,fv2] = fzero(@rate2,1)
   [x3,fv3] = fzero(@rate3,1)
   x2 = x2 * 12,x3      %比较 x2 和 x3 的值
```

## 实验九

1. (1) $\min z = -\frac{3}{4}x_1 + 150x_2 - \frac{1}{50}x_3 + 6x_4$

s. t. $\frac{1}{4}x_1 - 60x_2 - \frac{1}{50}x_3 + 9x_4 \leqslant 0$

$\frac{1}{2}x_1 - 90x_2 - \frac{1}{50}x_3 + 3x_4 \leqslant 0$

$x_1 \geqslant -5, x_2 \geqslant -5, -5 \leqslant x_3 \leqslant 1, x_4 \geqslant -5$

```
(2)c = [-3/4 150 -1/50 6];
A = [1/4 -60 -1/50 9;1/2 -90 -1/50 3];
```



```
b=[0;0];
Aeq=[ ]; beq=[ ];
vlb=[ -5; -5; -5; -5]; vub=[inf;inf;1;inf];
```

(3)

```
[x,fval]=linprog(c,A,b,Aeq,beq,vlb,vub)
```

2. (1)设生产甲产品 $x_1$ 个单位,生产乙产品 $x_2$ 个单位.则求解该问题的线性规划模型为:

$$\max \quad z=4x_1+3x_2$$

$$\text{s.t.} \quad \begin{cases} 2x_1+x_2\leqslant 10 \\ x_1+x_2\leqslant 8 \\ x_2\leqslant 7 \\ x_j\geqslant 0 \quad j=1,2 \end{cases}$$

(2)

```
c=[ -4 -3];
A=[2 1;1 1;0 1];
b=[10;8;7];
Aeq=[ ]; beq=[ ];
vlb=[0;0]; vub=[ ];
[x,fval]=linprog(c,A,b,Aeq,beq,vlb,vub)
```

3. (1) $\max f=21x_{11}+11x_{12}+x_{13}+31x_{21}+21x_{22}+11x_{23}+41x_{31}+31x_{32}+21x_{33}$

$x_{11}+x_{12}+x_{13}+x_{21}+x_{22}+x_{23}+x_{31}+x_{32}+x_{33}\leqslant 14\ 000$;

ABC 的买入量都小于 5 000

$x_{11}+x_{12}+x_{13}\leqslant 5\ 000, x_{21}+x_{22}+x_{23}\leqslant 5\ 000, x_{31}+x_{32}+x_{33}\leqslant 5\ 000$;

甲乙丙三种日需求总量分别为 3 000、2 000、1 000

$x_{11}+x_{21}+x_{31}=3\ 000, x_{12}+x_{22}+x_{32}=2\ 000, x_{13}+x_{23}+x_{33}=1\ 000$;

甲乙丙的辛烷值有要求:

$12x_{11}+6x_{21}+8x_{31}\geqslant 10(x_{11}+x_{21}+x_{31})$,
$12x_{12}+6x_{22}+8x_{32}\geqslant 8(x_{12}+x_{22}+x_{32})$,
$12x_{13}+6x_{23}+8x_{33}\geqslant 6(x_{13}+x_{23}+x_{33})$;

甲乙丙的硫含量有要求:

$0.5x_{11}+2.0x_{21}+3.0x_{31}\leqslant 1.0(x_{11}+x_{21}+x_{31})$,
$0.5x_{12}+2.0x_{22}+3.0x_{32}\leqslant 2.0(x_{12}+x_{22}+x_{32})$,
$0.5x_{13}+2.0x_{23}+3.0x_{33}\leqslant 1.0(x_{13}+x_{23}+x_{33})$;

加工汽油量为非负数

$x_{11}\geqslant 0, x_{12}\geqslant 0, x_{13}\geqslant 0, x_{21}\geqslant 0, x_{22}\geqslant 0, x_{23}\geqslant 0, x_{31}\geqslant 0, x_{32}\geqslant 0, x_{33}\geqslant 0$

运行如下程序:

```
c=[ -21 -11 -1 -31 -21 -11 -41 -31 -21];
```



```
a1 = [1,1,1,1,1,1,1,1,1
      1,1,1,0,0,0,0,0,0
      0,0,0,1,1,1,0,0,0
      0,0,0,0,0,0,1,1,1
      1,0,0,1,0,0,1,0,0
      0,1,0,0,1,0,0,1,0
      0,0,1,0,0,1,0,0,1
      -2,0,0,4,0,0,2,0,0
      0,-4,0,0,2,0,0,0,0
      0,0,-6,0,0,0,0,0,-2
      -0.5,0,0,1.0,0,0,2.0,0,0
      0,-1.5,0,0,0,0,0,1.0,0
      0,0,-0.5,0,0,1.0,0,0,2.0];
b1 = [14 000,5 000,5 000,5 000,3 000,2 000,1 000,0,0,0,0,0,0];
v1 = zeros(9,1);
[x,fval,exitflag,output,lambda] = linprog(c,a1,b1,[],[],v1,[],[])
          f = c * x
```

(2)

$$\max f = 21x_{11} + 11x_{12} + x_{13} + 31x_{21} + 21x_{22} + 11x_{23} + 41x_{31} + 31x_{32} + 21x_{33} - y_1 - y_2 - y_3$$

s. t.　　每天至多加工汽油总量为 14 000 桶

$$x_{11} + x_{12} + x_{13} + x_{21} + x_{22} + x_{23} + x_{31} + x_{32} + x_{33} \leqslant 14\ 000;$$

ABC 的买入量都小于 5 000

$$x_{11} + x_{12} + x_{13} \leqslant 5\ 000, x_{21} + x_{22} + x_{23} \leqslant 5\ 000, x_{31} + x_{32} + x_{33} \leqslant 5\ 000;$$

甲乙丙三种日需求量还要加上广告效应

$$x_{11} + x_{21} + x_{31} = 3\ 000 + 10y_1,$$
$$x_{12} + x_{22} + x_{32} = 2\ 000 + 10y_2,$$
$$x_{13} + x_{23} + x_{33} = 1\ 000 + 10y_3;$$

甲乙丙的辛烷值有要求:

$$12x_{11} + 6x_{21} + 8x_{31} \geqslant 10(x_{11} + x_{21} + x_{31}),$$
$$12x_{12} + 6x_{22} + 8x_{32} \geqslant 8(x_{12} + x_{22} + x_{32}),$$
$$12x_{13} + 6x_{23} + 8x_{33} \geqslant 6(x_{13} + x_{23} + x_{33});$$

甲乙丙的硫含量有要求:

$$0.5x_{11} + 2.0x_{21} + 3.0x_{31} \leqslant 1.0(x_{11} + x_{21} + x_{31}),$$
$$0.5x_{12} + 2.0x_{22} + 3.0x_{32} \leqslant 2.0(x_{12} + x_{22} + x_{32}),$$
$$0.5x_{13} + 2.0x_{23} + 3.0x_{33} \leqslant 1.0(x_{13} + x_{23} + x_{33});$$

加工汽油量为非负数,广告费也为非负数

$x_{11}\geqslant0, x_{12}\geqslant0, x_{13}\geqslant0, x_{21}\geqslant0, x_{22}\geqslant0, x_{23}\geqslant0, x_{31}\geqslant0, x_{32}\geqslant0, x_{33}\geqslant0$

$y_1\geqslant0, y_2\geqslant0, y_3\geqslant0$

运行如下程序:

```
c = [ -21  -11  -1  -31  -21  -11  -41  -31  -21 1 1 1];
a1 = [1,1,1,1,1,1,1,1,1,0,0,0
      1,1,1,0,0,0,0,0,0,0,0,0
      0,0,0,1,1,1,0,0,0,0,0,0
      0,0,0,0,0,0,1,1,1,0,0,0
      -2,0,0,4,0,0,2,0,0,0,0,0
      0,-4,0,0,2,0,0,0,0,0,0,0
      0,0,-6,0,0,0,0,0,-2,0,0,0
      -0.5,0,0,1.0,0,0,2.0,0,0,0,0,0
      0,-1.5,0,0,0,0,0,1.0,0,0,0,0
      0,0,-0.5,0,0,1.0,0,0,2.0,0,0,0];
b1 = [14 000,5 000,5 000,5 000,0,0,0,0,0,0];
a2 = [1,0,0,1,0,0,1,0,0,-10,0,0
      0,1,0,0,1,0,0,1,0,0,-10,0
      0,0,1,0,0,1,0,0,1,0,0,-10];
b2 = [3 000 2 000 1 000]';
v1 = zeros(12,1);
[x,fval,exitflag,output,lambda] = linprog(c,a1,b1,a2,b2,v1,[],[])
f = -c * x
```

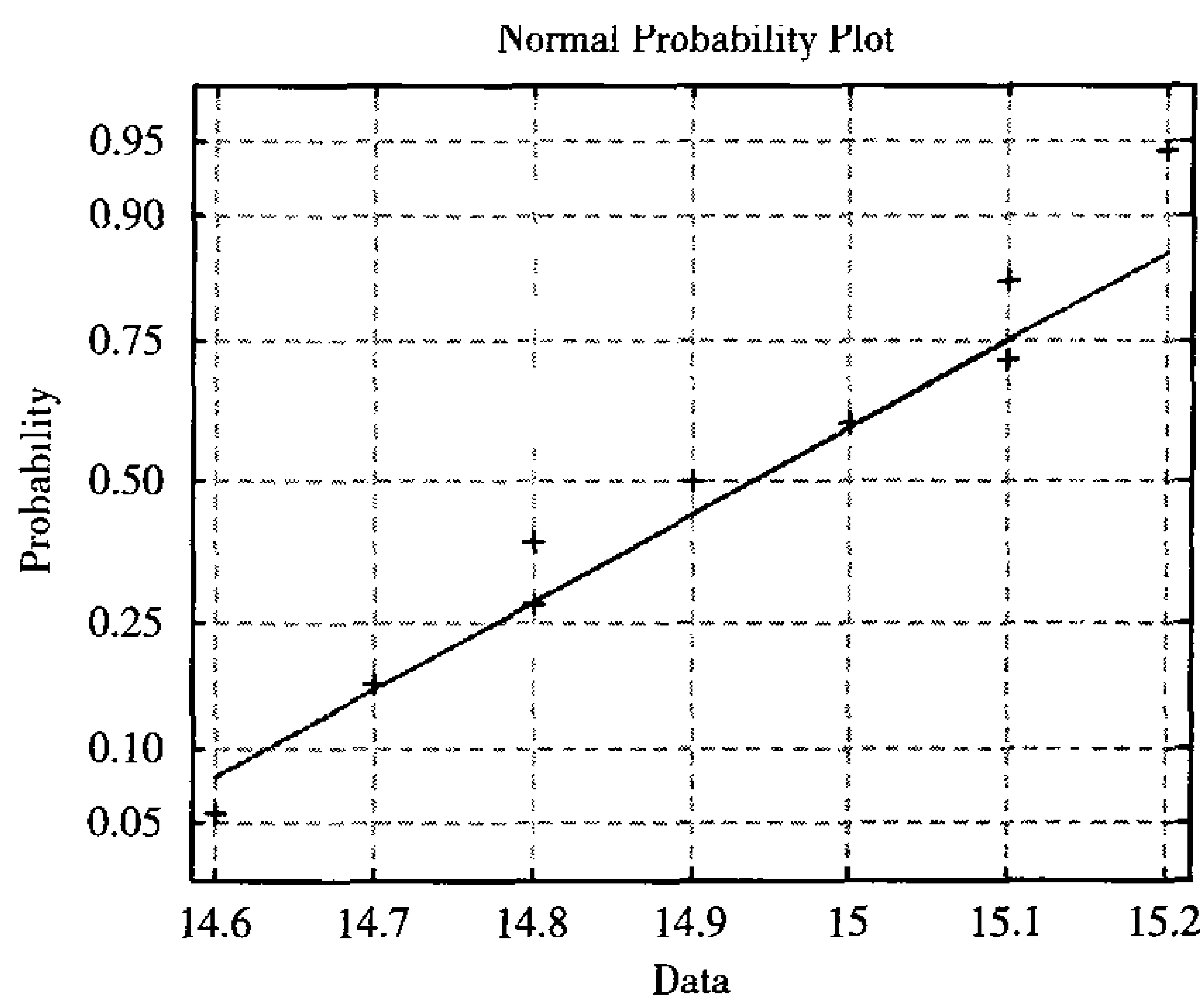
## 实验十

1. (3)
```
[max_temp,x] = max(temps)
```
(4)
```
[min_temp,x] = min(temps)
```
(5)
```
med_temp = median(temps)
avg_temp = mean(temps)
```
(6)
```
avg_avg = mean(avg_temp)
```
(7)
```
std_temp = std(temps)
```
(8)
```
corr_temp = corrcoef(temps)
```

2. (1)
```
x = [14.6 14.7 15.1 14.9 14.8 15.0 15.1 15.2 14.8];
alpha = 0.05;
[mu,sigma,muci,sigmaci] = normfit(x,alpha)
```

3. (1)
```
x = [14.6 14.7 15.1 14.9 14.8 15.0 15.1 15.2 14.8];
normplot(x)
```






结果如图所示,经观察这 9 个离散点非常靠近倾斜直线段,图形为线性的,因此可得出结论:该批滚珠的直径近似服从正态分布.

(2) h = ttest(x, 15,0. 05)

得:h = 0

检验结果:h = 0,表示不拒绝原假设,说明所提出的假设"直径均值为 15"是合理的.

h = ttest(x, 16,0. 05)

得:h = 1

检验结果:h = 1,表示拒绝原假设,说明所提出的假设"直径均值为 16"是不合理的.

## 实验十一

1. (1) xi = 0:0. 02:1;

```
yi = interp1(x, y, xi, 'linear');
zi = interp1(x, y, xi, 'spline');
wi = interp1(x, y, xi, 'cubic');
plot(x, y, 'o', xi, yi, 'r + ', xi, zi, 'g * ', xi, wi, 'k. - ')
legend('原始点', '线性点', '三次样条', '三次多项式')
```

(2) x = 0:0. 1:1;

```
y = [0. 3. 5 1 1. 4 1. 6 1. 6. 4. 8 1. 5 2];
xi = [ 0. 2500 0. 3500 0. 4500]
yi = interp1(x, y, xi, 'spline')
```

2. (1)

```
x = 0:0. 5:5;
y = 0:0. 5:6;
z = [89 90 87 85 92 91 96 93 90 87 82
     92 96 98 99 95 91 89 86 84 82 84
     96 98 95 92 90 88 85 84 83 81 85
```

```
    80 81 82 89 95 96 93 92 89 86 86
    82 85 87 98 99 96 97 88 85 82 83
    82 85 89 94 95 93 92 91 86 84 88
    88 92 93 94 95 89 87 86 83 81 92
    92 96 97 98 96 93 95 84 82 81 84
    85 85 81 82 80 80 81 85 90 93 95
    84 86 81 98 99 98 97 96 95 84 87
    80 81 85 82 83 84 87 90 95 86 88
    80 82 81 84 85 86 83 82 81 80 82
    87 88 89 98 99 97 96 98 94 92 87];
[X,Y] = meshgrid(x,y);
mesh(X,Y,z)  %绘原始数据图
(2)[xii,yii] = meshgrid(xi,yi);  %生成网格数据
   zii = interp2(x,y,z,xii,yii,'cubic');  %插值
   mesh(xii,yii,zii)  %加密后的地貌图
   hold on      % 保持图形
   [xx,yy] = meshgrid(x,y);  %生成网格数据
   plot3(xx,yy,z+0.1,'ob')  %原始数据用'O'绘出
3.(1)x=0:0.1:1;
     y=[.3.5 1 1.4 1.6 1.9.6.4.8 1.5 2]
     n=3;
     p=polyfit(x,y,n)
     xi=linspace(0,1,100);
     z=polyval(p,xi); %多项式求值
     plot(x,y,'o',xi,z,'k:',x,y,'b')
     legend('原始数据','3 阶曲线')
4.(1)function y=e11f1(a,x)
y=a(1)*x+a(2)*x.^2.*exp(-a(3)*x)+a(4);
(2)
x=0.1:0.1:1;
y=[2.3201,2.6470,2.9707,3.2885,3.6008,3.9090,4.2147,4.5191,4.8232,5.1275];
a=lsqcurvefit('e11f1',[1;2;2;3],x,y);
y1= e11f1 (a,x); plot(x,y,x,y1)
5.(1)x1=[0 34 67 101 135 202 259 336 404 471];
y1=[15.18 21.36 25.72 32.29 34.03 39.45 43.15 43.46 40.83 30.75];
x2=[0 24 49 73 98 147 196 245 294 342];
y2=[33.46 32.47 36.06 37.96 41.04 40.09 41.26 42.17 40.36 42.73];
plot(x1,y1);figure;
plot(x2,y2);
```

分析:使用 MATLAB 语言首先画出土豆产量与氮施肥量的散点图,从图可看出土豆产量与氮肥量的关系是二次函数关系,因此可选取拟合函数为:

$$y = ax^2 + bx + c$$

其中 x 和 y 分别为氮肥量和土豆产量,a、b 和 c 为待定系数.再画出磷肥量与土豆产量的散点图,从图可看出从 0 到 98、从 98 到 342 之间分别呈明显的线性关系.由此可选取所求拟合函数为一分段的线性函数,换言之,用前 5 点作一线性拟合函数,再用后 6 个点也作一线性拟合函数,最后用两个线性函数求出其分界点即可得分段线性函数.

(2)
```
x1 = [0, 34, 67, 101, 135, 202, 259, 336, 404, 471];
y1 = [15.18, 21.36, 25.72, 32.29, 34.03, 39.45, 43.15, 43.46, 40.83, 30.75];
plot(x1, y1, 'r +')
aa = polyfit(x1, y1, 2)
xx = 0:471;
yy = aa(1) * xx. * xx + aa(2) * xx + aa(3);
plot(xx, yy, x1, y1, 'r +')
% Next problem
x2 = [0, 24, 49, 73, 98, 147, 196, 245, 294, 342];
y2 = [33.46, 32.47, 36.06, 37.96, 41.04, 40.09, 41.26, 42.17, 40.36, 42.73];
plot(x2, y2, 'r +')
a1 = polyfit(x2(1:5), y2(1:5), 1)
a2 = polyfit(x2(5:10), y2(5:10), 1)
x0 = (a2(2) - a1(2))/(a1(1) - a2(1))
xx1 = 0:x0; yy1 = a1(1) * xx1 + a1(2);
xx2 = x0:342; yy2 = a2(1) * xx2 + a2(2);
plot(x2, y2, 'r +', xx1, yy1, xx2, yy2)
```

可得:对氮肥的拟合函数为:

$$y = -0.000339532x^2 + 0.197150x + 14.7416$$

对磷肥的拟合函数为:

$$y = \begin{cases} 0.0844453x + 32.0771 & 0 \leq x \leq 100.507 \\ 0.00592986x + 39.9685 & 100.507 < x \leq 342 \end{cases}$$

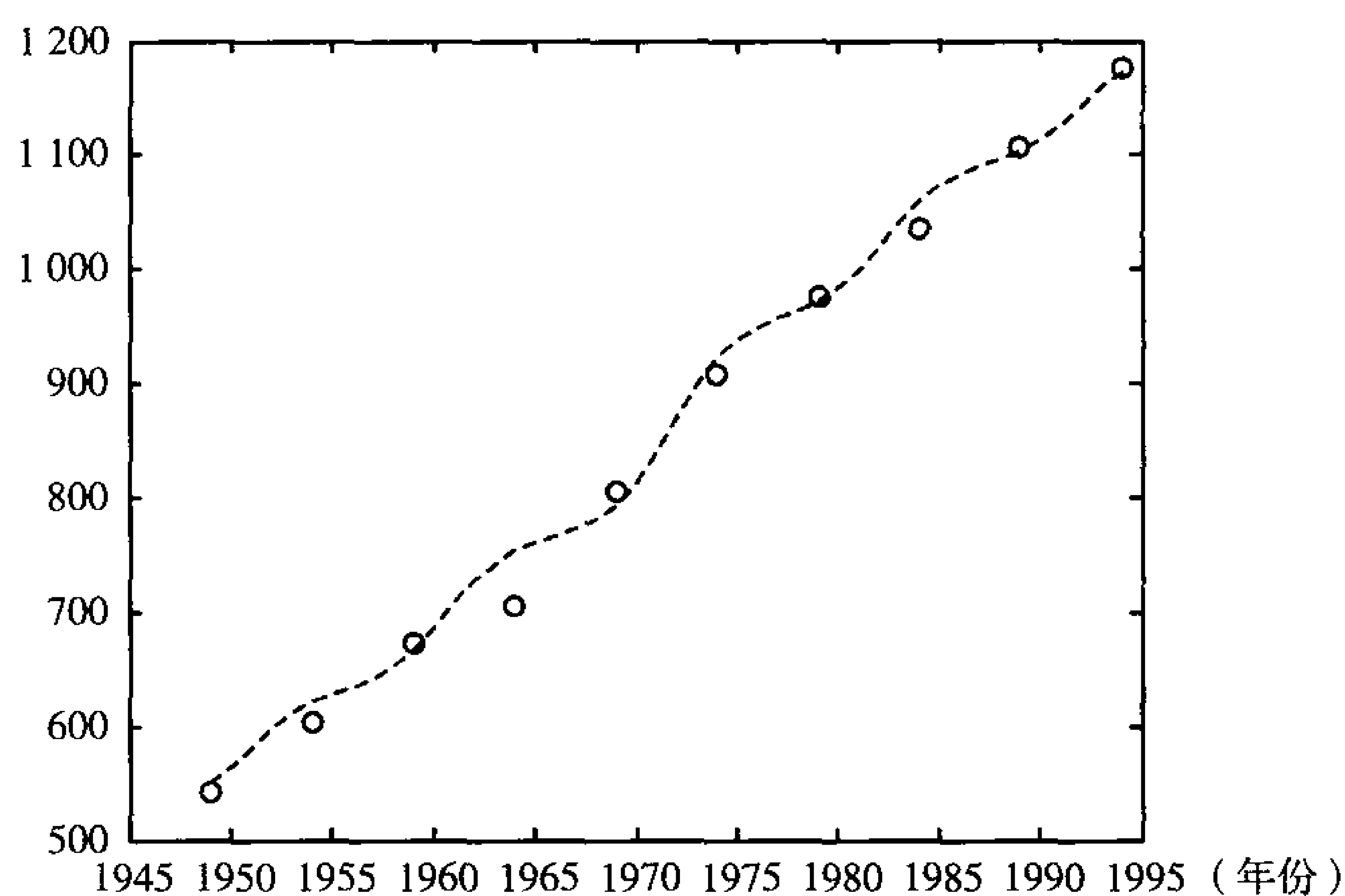
## 实验十二

1. (1)
```
x = 0:0.1:1;
y = [0.3 0.5 1 1.4 1.6 1.9 0.6 0.4 0.8 1.5 2];
x0 = 0:0.01:1;
net = newff(x, y, [5, 1], {'tansig', 'tansig'});
% net.trainParam.max_fail = 10;
% net.trainParam.min_grad = 1e-20;
```



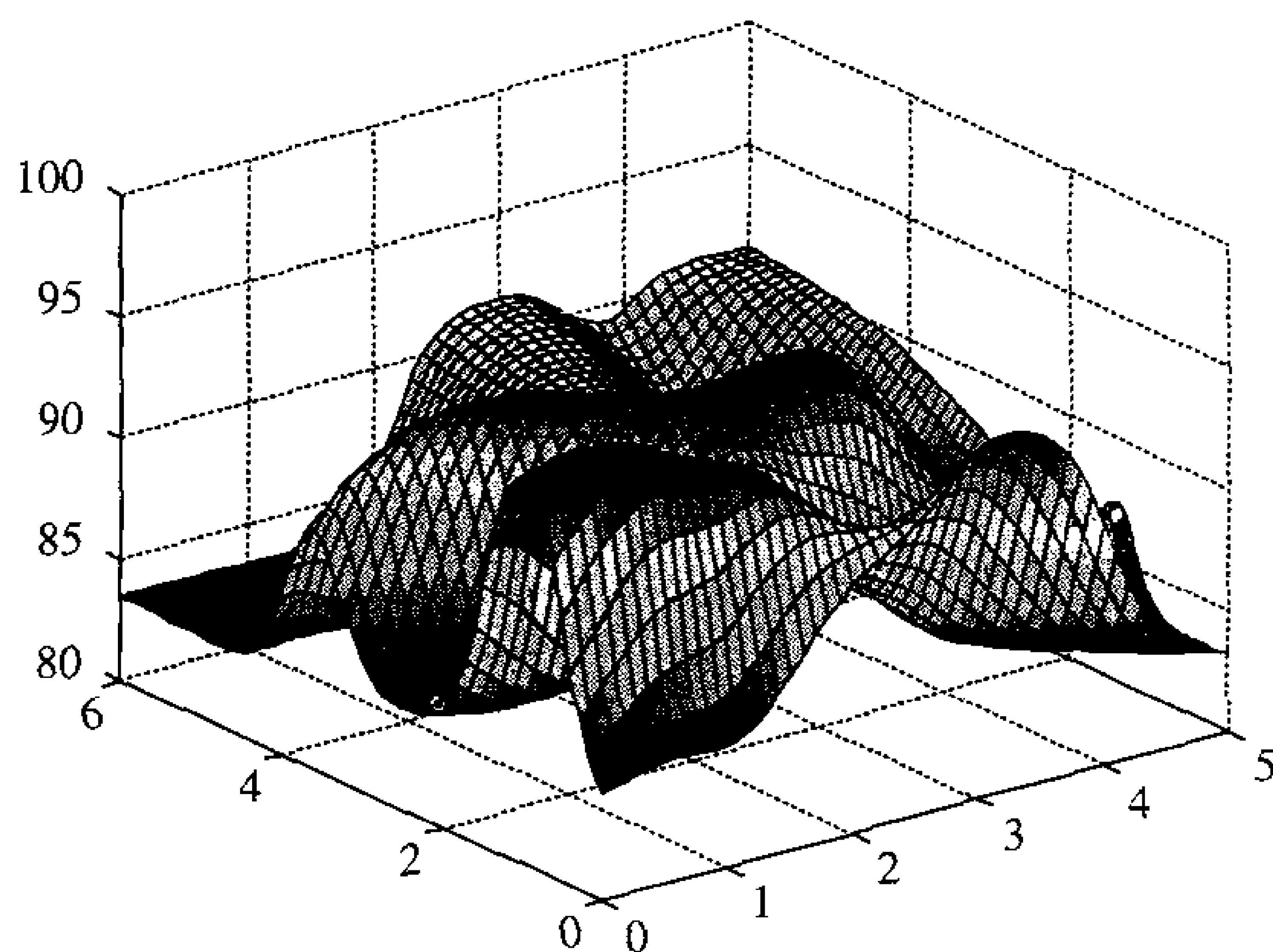
```
net = train(net, x, y);
y0 = sim(net, x0) ;
figure, plot(x, y, 'o', x0, y0, ':r') ;
```

2.
```
x = 1949:5:1994;
y = [541.67 602.66 672.09 704.99 806.71 908.59 975.42 1 034.75 1 106.76 1 176.74];
x0 = 1949:1994;
net = newff(x, y, [5, 1], {'tansig', 'tansig'});
% net. trainParam. max_fail = 10;
% net. trainParam. min_grad = 1e - 20;
net = train(net, x, y);
y0 = sim(net, x0) ;
figure, plot(x, y, 'o', x0, y0, ':r') ;
```




3.
```
[x, y] = meshgrid(0:0.5:5, 0:0.5:6);
z = [89 90 87 85 92 91 96 93 90 87 82
     92 96 98 99 95 91 89 86 84 82 84
     96 98 95 92 90 88 85 84 83 81 85
     80 81 82 89 95 96 93 92 89 86 86
     82 85 87 98 99 96 97 88 85 82 83
     82 85 89 94 95 93 92 91 86 84 88
     88 92 93 94 95 89 87 86 83 81 92
     92 96 97 98 96 93 95 84 82 81 84
     85 85 81 82 80 80 81 85 90 93 95
     84 86 81 98 99 98 97 96 95 84 87
     80 81 85 82 83 84 87 90 95 86 88
     80 82 81 84 85 86 83 82 81 80 82
     87 88 89 98 99 97 96 98 94 92 87];
x = x(:)'; y = y(:)'; z = z(:)';   % 注意这三个向量为行向量
net = newff([x; y], z, [10, 10, 1], {'tansig', 'tansig', 'tansig'});
```

```
net.trainParam.epochs = 1 000; net.trainFcn = 'trainlm';
[net,b] = train(net,[x; y],z);   % 训练神经网络
xi = linspace(0,5,50);   %加密横坐标数据到 50 个
yi = linspace(0,6,80);   %加密纵坐标数据到 60 个
[x2,y2] = meshgrid(xi,yi); x1 = x2(:)'; y1 = y2(:)';
figure; z1 = sim(net,[x1; y1]); z2 = reshape(z1,size(x2));
mesh(x2,y2,z2)
```




## 实验十三

```
(1)p = [2.00 2.50 3.00 3.50 4.00 4.50 5.00 5.50 6.00];
l = [41 38 34 32 29 28 25 22 20];
ad = [0 10 20 30 40 50 60 70];
lf = [1.00 1.40 1.70 1.85 1.95 2.00 1.95 1.80];
plot(p,l);
figure,plot(ad,lf);
```

由结果可看出,销售价格与销售量的函数关系可以一次多项式来拟合,广告费用与销售增长因子的函数关系可用二次多项式来拟合.

```
(2)pp1 = polyfit(p,l,1)
pp2 = polyfit(ad,lf,2)
pp1 =
    -5.1333      50.4222
pp2 =
    -0.0004      0.0409      1.0188
```

(3)利润 P = 销售收入 - 成本 - 广告费用

P = (x - 2) * l * lf - y

将步骤 2 中所得的售价与销量的函数关系,以及广告费与销量增长因子的关系代入以上关系式可得:

P = (x - 2) * ( - 5.1333 * x + 50.4222) * ( - 0.0004 * y^2 + 0.0409 * y + 1.0188) - y

(4)建立存储目标函数的 pfun. m 文件:

```
function fx = pfun(x)
fx = -(x(1) -2) * ( -5. 1333 * x(1) +50. 4222) * ( -0. 0004 * x(2)^2 +0. 0409 * x(2) +
1. 0188) + x(2);
```

调用 fminunc()求解目标函数的最小值.

```
x0 = [1, 10];[x, fval] = fminunc('pfun', x0)
```

可得:x =

```
    5. 9113    35. 2074
fval =
    -118. 9437
```

可知,当售价为 5. 9113 元,广告投入为 35 207. 4 元时,利润最大,最大利润为118 943. 7元.

## 实验十四

(1)由牛顿第二定律可列出圆桶下沉速度的微分方程:

$$\frac{dv}{dt}=\frac{W-B-f}{m}=\frac{W-B-Cv}{m}$$

其中 W 为圆桶重量,B 为浮力,C 为下沉阻力与速度关系的比例系数.

```
clear;
W =239. 46 * 9. 8;% 圆桶所受重力(N)
B =213. 528 * 9. 8;% 浮力(N)
m =239. 46;% 圆桶的质量(kg)
C =0. 19 * 9. 8% 比例系数(Ns/m)
```

求解微分方程的解析解:

```
v = dsolve('Dv = (2345. 7 -2092. 6 - 1. 862 * v)/239. 46', 'v(0) =0', 't');
```

得:

```
v =
    126550/931 - 126550/931 * exp( -931/119730 * t)
```

(2)s = int(v)

```
s =
    126550/931 * t + 15151831500/866761 * exp( -931/119730 * t)
```

(3) vt = limit(v, + inf)

vt = 135. 9291 >12. 2m/s,因此仍不能判断圆桶到底海底时是否会破裂.

(4)vx = dsolve('Dv = (2345. 7 -2092. 6 - 1. 862 * v)/239. 46 * v', 'v(0) = 0', 'x');

```
vx =
    1/11973 * 303036630^(1/2) * x^(1/2)
    -1/11973 * 303036630^(1/2) * x^(1/2)
```

选择 vx =1/11973 * 303036630^(1/2) * x^(1/2)

```
subs(vx, x, 91. 14)
```



```
ans =
    13.8803
```

可知当圆桶沉到海底时,速度超过 12.2m/s,因此生态学家和工程师是有道理的.

(5)

```
clear;
h =0.1;% 时间点间隔
ts =[0:h:2000];% 大致时间取值范围
x0 =0;% 初始条件
[t,x] =ode45(@sd,ts,x0,[],G,F,m,b);% ode45 计算微分方程数值解
plot(t,x,'-'),grid% 输出 v(t)图形
xlabel('t');
ylabel('v(t)');
```



# 参考文献

1. 刘卫国．MATLAB 程序设计教程．北京：中国水利水电出版社．2005.

2. 张志涌．MATLAB 教程．北京：北京航空航天大学出版社．2006.

3. 薛定宇，陈阳泉．高等应用数学问题的 MATLAB 求解．北京：清华大学出版社．2004.

4. 万福永．戴洪晖等．数学实验教程．北京：科学出版社．2006.

5. 李继成．数学实验．北京：高等教育出版社．2006.